

Publicación práctica para usuarios de

commodore

Revista mensual 1986 Precio 350 Ptas Año 1 Número 9



CONTROL MULTITECLA



PROGRAMA TRAZADOR

COMO FUNCIONA UN MONITOR



AHORA TE PUEDES PROGRAMAR UN VIAJE ALUCINANTE AL EPCOT CENTER CON ALEA. ES LOGICO.

Te presentamos, en estreno mundial, todo un reto a tu inteligencia: la colección de programas Logicolor.

Con los tres juegos de la colección Logicolor tu mente desafiará a la fría lógica del ordenador. ¡Atrévete con ellos!

AUTOS LOCOS: Construye tu propia escudería y apuesta por tu bólido favorito. Un primer contacto con el uso de los símbolos. Para chicos entre 10 y 12 años. Incluye también un super-master mind contra el ordenador.

MANZANAS Y GUSANOS: Utilizando fórmulas puedes recoger las manzanas y dejar fuera los gusanos; proteger las ánforas de los golpes del martillo; defender los globos aerostáticos de las flechas enemigas; o evitar que los cañones destruyan las torres de tu fortaleza. Tu inteligencia lógica es la única arma que necesitas. Para chicos entre 12 y 14 años, y para quienes desean mantener su mente en forma.

REHENES: Tendrás que desarrollar una estrategia lógica si quieres eliminar a los conspiradores y salvar la corona. ¿Te gustaría descubrir la fórmula que abre el cofre de los diamantes? Intenta descubrir un procedimiento lógico para rescatar a los rehenes. Para chicos entre 14 y 16 años, y para quienes se las dan de genios.

Además, la compra de cada programa de la colección Logicolor te da derecho a participar en el fabuloso concurso EPCOT, y si consigues vencer al ordenador, tus posibilidades de conseguir un magnífico premio se duplican.

Si resultas ganador puedes elegir uno de estos SUPERPREMIOS:

1) UN FANTASTICO VIAJE PARA DOS PERSONAS DE 9 DIAS AL EPCOT CENTER; visitarás Marineland, el Museo Aeroespacial de la Nasa, Disnyeworld, el Epcot Center y otros muchos lugares.
2) UN SUPERORDENADOR IBM-PC portátil.
3) UNA PAGA MENSUAL DE 30.000 Pts. durante un año para ti solo.

Encontrarás las Bases para participar en el concurso, junto con las fichas, en cada programa. Envíalas a ALEA antes del 21 de Julio de 1986.



¡Atención!, si envías tus fichas antes del 23 de Junio, tus posibilidades de ganar son aún mayores.

Alea también ha pensado en los más "peques", tus hermanos de 4 a 9 años. Para ellos tenemos una serie de juegos que les ayudarán al aprendizaje de la escritura y la lectura. Comprando cualquiera de ellos, participarás automáticamente en el concurso LEXA, pudiendo llegar a conseguir una beca de estudios de hasta 500.000 Pts.

Puedes pedir tus programas llamando al teléfono de Madrid (91) 446 57 64 o bien enviándonos el cupón que hay al pie de esta página. También encontrarás los programas de la colección Logicolor en la microtienda de tu barrio, El Corte Inglés y Galerías Preciados.

Animo, por sólo 3.875 Pts. obtienes un magnífico programa y ¡hasta cuatro participaciones para el gran concurso EPCOT!

¡NO LO DEJES ESCAPAR! Prográmate ahora mismo un premio alucinante.



apartado de correos 10.048 de Madrid

GRUPO SOFT



TCA

INPUT COMMODORE

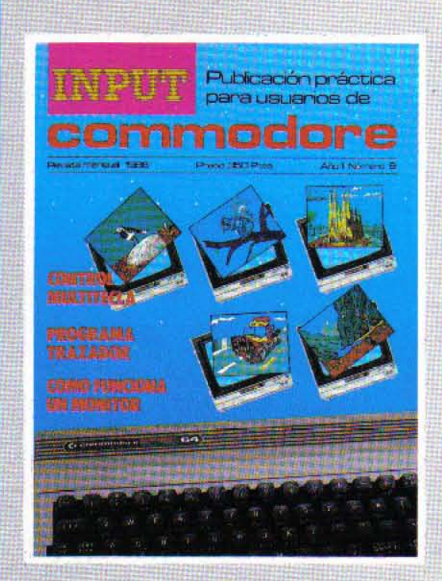
		COMMODORE 64/128	MSX/64K	SPECTRUM 48/PLUS	AMSTRAD 464/664/6128
COLECCION LEXA	EL DUENDE				
	EL TESORO				
	EL TORREON				
	EL OASIS				
COLECCION LOGICOLOR	AUTOS LOCOS				
	MANZANAS Y GUSANOS				
	REHENES				

Deseo adquirir los siguientes programas de su biblioteca, al precio de 3.875 Pts. cada uno, más 465 Pts. IVA.
Con la compra de los programas adquiero el derecho a participar en el concurso EPCOT y/o LEXA.

Forma de pago:
☐ Contrarreembolso
☐ Con cheque adjunto a nombre de Alea. S. A.
☐ Con cargo a la tarjeta de crédito:
☐ Visa
☐ American Express
☐ Dinners Club

Nombre del titular: ______
N.° de la tarjeta: ______
Válida desde ___/___ hasta ___/___
Firma del titular:

Nombre: ______
Dirección: ______
Tel.: ______



AÑO 1 NUMERO 9

DIRECTOR:
Alejandro Diges
COORDINADOR EDITORIAL:
Francisco de Molina
DISEÑO GRAFICO:
Tomás López
COLABORADORES:
Luis R. Palencia, Christophe Pais, Francisco Tórtola, Benito Román, Esther de la Cal.

INPUT Commodore es una publicación juvenil de EDICIONES FORUM

GERENTE DIVISION DE REVISTAS:
Angel Sabat

PUBLICIDAD: José Real-Grupo Jota
Madrid: c/ General Varela, 35
Teléf. 270 47 02/03
Barcelona: Avda. de Sarriá, 11-13, 1.º
Teléf. 250 23 99

FOTOMECANICA: Ochoa, S. A.

COMPOSICION: EFCA, S. A.

IMPRESION: Edime, S. A.
Depósito legal: M. 27.884-1985

SUSCRIPCIONES: EDISA,
López de Hoyos, 141. 28002 Madrid
Telef. (91) 415 97 12

REDACCION:
Alberto Alcocer, 46, 4.º
28016 Madrid. Teléf. 250 10 00

DISTRIBUIDORA
R.B.A. PROMOTORA DE EDICIONES, S. A.
Travesera de Gracia, 56. Edificio Odiseus.
08006 Barcelona
El precio será el mismo para Canarias que para la Península y en él irá incluida la sobretasa aérea.

Se ha solicitado el control OJD

INPUT Commodore es independiente y no está vinculada a Commodore Business Machines o sus distribuidores.

INPUT no mantiene correspondencia con sus lectores, si bien la recibe, no responsabilizándose de su pérdida o extravío. Las respuestas se canalizarán a través de las secciones adecuadas en estas páginas.

INPUT commodore

SUMARIO

UN MUNDO MUY MOVIDO

Aunque eran de sobra conocidas las dificultades financieras por las que venía pasando durante el último año **Sinclair Research,** a todos sorprendió la noticia de la compra de ésta firma por parte de su más directo rival: **Amstrad.** Se da la paradoja de que además la firma fue adquirida a precio de saldo, sin contar las deudas contraídas con los proveedores.

Alan Sugar es el artífice del *boom* de **Amstrad,** un hombre conocedor de los mercados internacionales de componentes, en los que se mueve ágilmente desde que lanzase sus primeros equipos de alta fidelidad. Ambos personajes tiene algo más en común que ser británicos, comparten la idea fija de poner los avances de la tecnología electrónica al alcance de cualquier consumidor.

Sin embargo, mientras que *sir* **Clive** tiene vocación investigadora, **Sugar** es un experto en el control de compras y *stocks,* factor este último que llevo a **Sinclair** a la bancarrota, al acumularse una gran cantidad de equipos invendidos en sus almacenes. Es curioso observar como la compañía que no ha mucho estuvo valorada en 136 millones de libras esterlinas, haya sido comprada por tan sólo 5.

Por otro lado ambas firmas parece que disponen de modelos muy avanzados de máquinas dispuestas a enfrentarse en las tiendas con los **Amiga,** de **Commodore,** y **520 ST,** de **Atari** e incluso los atractivos **MSX II.** Este tipo de competencia tan directa promete ser estimulante para el mercado y la producción de nuevo *software.* Pero sin duda los más beneficiados deberán ser los usuarios que accedan a la nueva generación de microordenadores con elevadas prestaciones.

Algunos de los que se incorporan a nuestro inmejorable conjunto de lectores, escriben a la redacción para expresarnos su desconcierto por la paginación del artículo que ocupa las ocho páginas centrales de INPUT. Piensan que su ejemplar está mal encuadernado o que nos hemos confundido al hacer la revista. Nada más lejos de la verdad para su tranquilidad. Se trata de un curso de aprendizaje del BASIC —código máquina en el futuro— a través de un coleccionable, que explica el modo de desarrollar juegos propios. Dentro de algunos números lanzaremos unas tapas que permitirán que este coleccionable se transforme en un libro y, por tanto, cada capítulo no tiene porqué ocupar las ocho páginas, pudiendo ser más o menos.

LOS MEJORES DE INPUT

Hemos pensado que es interesante disponer de un ***ranking*** que ponga en claro, mes a mes, cuáles son los programas preferidos de nuestros lectores. Para ello, es obligado preguntaros directamente y tener así el mejor termómetro para conocer vuestras preferencias. Podéis votar por cualquier programa aunque no haya sido comentado todavía en **INPUT**.

El resultado de las votaciones será publicado en cada número de **INPUT**.

Entre los votantes sortearemos 10 cintas de los títulos que pidáis en vuestros cupones.

Nota: No es preciso que cortéis la revista, una copia hecha a máquina o una simple fotocopia sirven.

Enviad vuestros votos a: **LOS MEJORES DE INPUT** Alberto Alcocer, 46 - 4.° B. 28016 Madrid

ELIGE TUS PROGRAMAS

Primer título elegido ____________________

Segundo título elegido ____________________

Tercer título elegido ____________________

Programa que te gustaría conseguir ____________________

Qué ordenador tienes ____________________

Nombre ____________________

1.er Apellido ____________________

2.° Apellido ____________________

Fecha de nacimiento ____________________

Teléfono ____________________

Dirección ____________________

Localidad ____________________

Provincia ____________________

INPUT COMMODORE N.° 9



AMSTRAD SIGUE IMPACTANDO.

Imparable, Amstrad sigue ganado puestos, hasta tal punto que se estima puedan haberse vendido casi cincuenta mil unidades del CPC 6128, previéndose un fuerte incremento en base al menor P.V.P., unas ochenta y cinco mil pts. en la versión con fósforo verde y cerca de ciento veinte mil si es de color, IVA aparte, por supuesto.
Otra baza que parece estar jugando este fabricante es la del sistema operativo CP/M, de Digital Research, que aún no siendo precisamente de nuevo cuño, sí dispone de una gran biblioteca de programas. Este mismo sistema está siendo utilizado por Commodore en el C-128, aunque ya existía en cartucho para el C-64, y se especula en torno a que Sinclair lo utilizará en breve (curiosa coincidencia con la reciente absorción de Sinclair).
Una de las dificultades que surgen con el CP/M es la falta de compatibilidad con el formato del *diskette*. A este respecto, el CPC 6128 dispone ya de programas profesionales tan populares como el Multiplan (potente hoja de trabajo), el SuperCalc II (otra popular hoja), el dBase II (una de la bases de datos más extendidas), etc..



LOS 10 HITS EN UN PAQUETE

Gallup es un prestigioso y conocido instituto que, entre otras cosas, se dedica a esclarecer las cifras reales que en el mundo se barajan.
La alianza Gallup/Microscope elabora desde hace algun tiempo una serie de tablas, que recogen el volumen de ventas y otras informaciones contrastadas con las obtenidas a partir de los profesionales- e incluso las propias revistas- de los programas existentes en el mercado britanico. De esta manera se pretende evitar las especulaciones. Dichos resultados son regularmente publicados por diversas revistas britanicas también.
Ahora, ABC Soft tiene intención de poner a la venta mensualmente la recopilación de los diez programas con mayor éxito de ventas, segun las evoluciones semanales del mencionado *ranking*. La selección es realizada por la firma Beau-Jolly.



MAS CONFORT EN LA PANTALLA

Essilor es una importante firma francesa de la industria óptica. Durante algún tiempo sus técnicos han estado desarrollando lo que han dado en definir como "una lente antichoque, dos veces más ligera que la lente mineral".
Este producto comercial, Essilor Soft, ha sido desarrolado para proteger la vista de quienes trabajan habitualmente con monitores de vídeo.
Su función consiste en evitar el deslumbramiento y los reflejos, aumentando asimismo el contraste de la imagen.
Para presentar este avance de la óptica, fue convocado un panel de expertos en distintas áreas que ofrecieron una panorámica más clara de los problemas visuales derivados del trabajo con pantallas conectadas a ordenador y sus posibles compensaciones.



CURSO DE BASIC

Ceac es un centro de enseñanza multidisciplinaria a distancia. Uno de sus productos mas recientes es un curso de lenguaje BASIC con profusión de ejemplos, que comienza desde el nivel mas bajo.
Por otro lado, los cursos han sido escritos en versiones personalizadas para Commodore y Spectrum.
Cada lección incorpora un cuestionario de evaluación, que es corregido en la sede del centro.



EL AMIGA, POR FIN

Fuentes internas de la firma Microelectrónica y Control han afirmado a INPUT que será posible ver el controvertido Amiga en su *stand* del próximo salón Informat, que se celebra durante el presente mes en Barcelona.
Este avanzado modelo de Commodore sigue suscitando el interés de los usuarios de la marca, aunque los programadores estadounidenses siguen achacándole gran complejidad en la programación.



PC AMPLIADO

El PC de Commodore se reconfigura en la nueva versión II. A partir de ahora, la memoria RAM mínima asciende hasta los 512 Kbytes, incluyéndose una potente tarjeta gráfica, que anteriormente era opcional.
Por otro lado, el modelo PC 20 ve ampliada su capacidad en disco duro hasta 20 Mbytes.
Los nuevos toques de personalidad vienen por parte del teclado y el sistema operativo, que han sido castellanizados.

ERBEdades



"V"

La serie de ciencia-ficción televisiva invade tu ordenador. La Tierra es visitada por alienígenas de aspecto humano. Si embargo, estos reptiles pronto revelarán su objetivo final... conquistar la Tierra y esclavizar a la raza humana.

GREEN BERET

Green Beret (boina verde), es el programa de mayor acción que hayas visto jamás en un ordenador.
Decir Green Beret es decir un hombre perfectamente entrenado para la lucha. Prepárate...

THE WAY OF THE TIGER

Entra en el mudo de los samurais. Mantén la calma mientras el movimiento y las rutinas de combate te transportan a niveles que nunca pensaste posibles. Experimenta los sorprendentes efectos del "Triple Scroll" mientras intentas mejorar tus técnicas de lucha cuerpo a cuerpo, con espada samurai o con mil posibilidades más.



DESERT FOX

¡La guerra ha estallado en Africa! Las condiciones de lucha son infrahumanas. El calor agobiante y el mariscal Rommel hacen imposible la supervivencia del ejército aliado. Tú eres un lobo solitario, tu enemigo es el zorro del desierto. El mejor simulador de guerra está en tus manos!

URIDIUM

¡¡El sistema solar está siendo atacado!! Las estaciones interespaciales enemigas han sido colocadas en órbita alrededor de nuestro sistema y están absorbiendo los recursos minerales. Con tu nave espacial tipo "Manta" serás transportado a cada planeta para destruir las bases enemigas. "Uridium" es el juego de acción más rápido y adictivo que puedas encontrar.

URIDIUM
HEWSON
ERBE


PING PONG
ERBE
Imagine
Konami
PING-PONG
La gran sorpresa. Gráficos increíbles, movimiento super-rápido, podrás efectuar las mismas jugadas que si tuvieras la paleta en tu mano.
Botes, rebotes, efectos, dejadas, saques, cortadas, mates... todo es posible con esta maravilla llamada "Ping-Pong".
COMMODORE
M
Melbourne House
ROCK'N LUCHA
ERBE Software
ROCK'N LUCHA
El primer juego de lucha libre hecho para ordenador. Más de 25 movimientos diferentes te permitirán hacer todo tipo de llaves, desde la sujección de espaldas, hasta la voltereta con los hombros, pasando por el rebote en las cuerdas del ring. Hecho por los mismos programadores del legendario "Exploding-Fist".
U.S. GOLD
KUNG-FU MASTER
THE OFFICIAL COIN OP GAME
KUNG-FU MASTER
El juego de Kárate que ha sido número 1 en U.S.A. durante 3 meses. El auténtico programa de las máquinas recreativas que siempre has deseado tener.

GANADOR DEL SORTEO

ENCUESTA: "AYUDANOS A AYUDARTE"

Vuestra respuesta ha sido masiva y ya estamos procesando los datos para conocer con mayor precision los gustos del lector medio de INPUT. Desde aquí os damos las gracias por vuestra inestimable colaboración.

Nos habría gustado daros un premio a todos. Sin embargo los invisibles dedos de la fortuna han sacado este nombre del montón de respuestas:



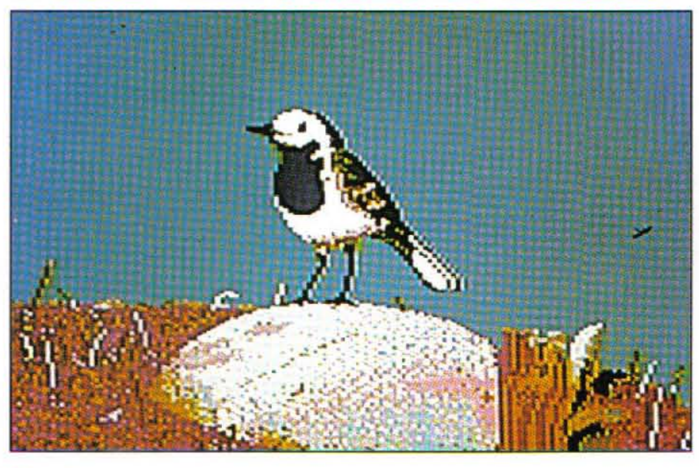
ELISABETH GUIXE CALDERON
de Barcelona.

Quien recibirá el prometido COMMODORE 64 a lo largo de este mes. Enhorabuena.

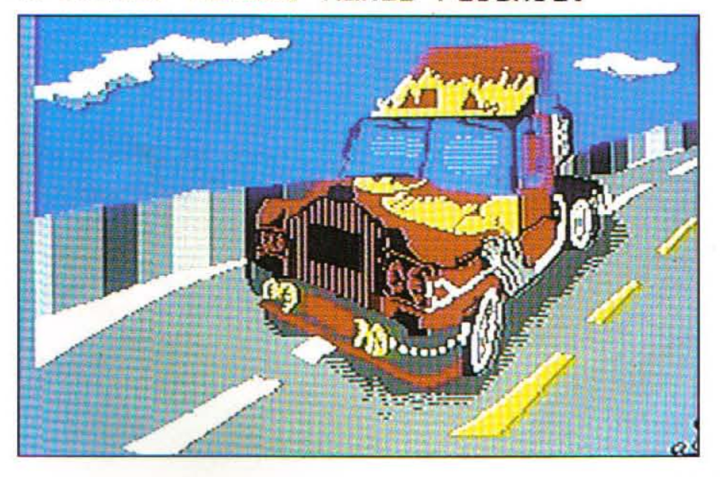
GANADORES:
CONCURSO DE ARTE POR ORDENADOR

Lo habeis puesto muy difícil. Estos son los trabajos premiados tras la discutida votación de los miembros del jurado.
A lo largo de este mes recibireis los premios ofrecidos por Microbyte.
Finalmente se decidió premiar con quince suscripciones por un año a otros tantos concursantes, cuya relación publicaremos en el próximo numero.

1er.Premio: *Pajarita de las nieves.*
Artista: Adolfo Gutiérrez González.



2º.Premio: *Camión.*
Artista: Rafael Muñoz Polonio.



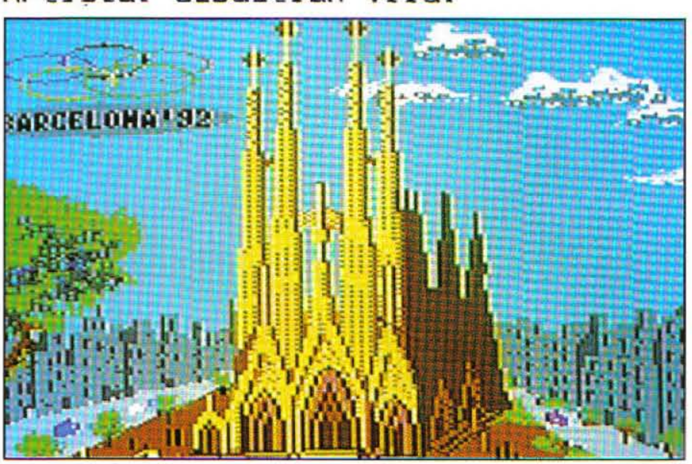
3er.Premio: *Marilin.*
Artista: J. Luis Berdún Teruel.



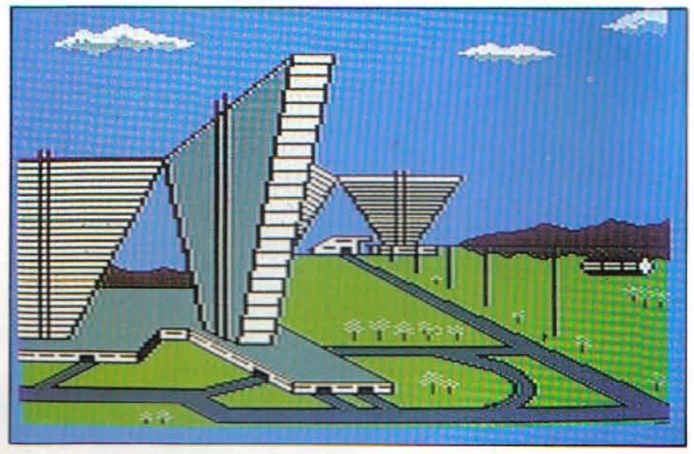
1er.Premio: *Ardilla.*
Artista: Carlos Vilasís Faura.



2º.Premio: *Barcelona.*
Artista: Sebastían Vila.



3er.Premio: *Ciudad Futurista.*
Artista: Agustín Ferreiro Carnota.





¡NO TE PIERDAS NI UN SOLO EJEMPLAR!

INPUT COMMODORE quiere proporcionar a sus lectores este nuevo servicio de ejemplares atrasados para que no pierdan la oportunidad de tener en sus hogares todos los ejemplares de esta revista, líder en el mercado español.

Podréis solicitar cualquier número de **INPUT COMMODORE** que queraís, siempre al precio de cubierta (sin más gastos).

Utiliza el cupón adjunto, enviándolo a **EDISA** (Dpto. de Suscripciones), López de Hoyos, 141 - 28002 Madrid, o bien llámanos por teléfono al (91) 415 97 12.





siempre a tu servicio

CUPON DE PEDIDO

SI, envíenme contrarreembolso .... ejemplares de **INPUT COMMODORE** de los números:

(marca con una (X) tu elección)

[1] [2] [3] [4] [5] [6] [7] [8]

NOMBRE

APELLIDOS

DOMICILIO

NUM. PISO ESCALERA COD. POSTAL

POBLACION PROV.

TELEFONO FIRMA

CONTROL MULTITECLA

- COMO FUNCIONA TU TECLADO
- DIFERENTES METODOS
- IMPLEMENTACION DEL CONTROL DE JUEGOS
- AUTORREPETICION

Si quieres tener un control máximo cuando juegues, en vez de utilizar un *joystick* puedes programar tu teclado para detectar más de una pulsación de tecla.

Una de las características más importantes de cualquier juego es el tipo de interacción existente entre el ordenador y el usuario. Para la mayoría de los fines el *joystick* ofrece un control más sofisticado, aunque probablemente es más versátil el teclado. Para saber qué tecla se ha pulsado, se utiliza normalmente la sentencia GET.

GET es capaz de detectar casi todos los caracteres, pero su principal desventaja es que sólo puede detectar dos pulsaciones simultáneas cuando una de las teclas es SHIFT u otra tecla de control análoga, tal como CTRL.

La detección de estas combinaciones es adecuada para casi todas las aplicaciones, pero surgen problemas cuando, por ejemplo, quieres controlar el movimiento vertical y horizontal, disparar el láser y arrojar bombas inteligentes sobre un avión. El conjunto de todas estas actuaciones requiere la pulsación simultánea de cuatro teclas. La mejor manera de resolver el problema, aunque de hecho puede que no siempre sea posible, depende de cómo esté hecho el ordenador.

EFECTO DE LAS PULSACIONES DE TECLA

Para entender la manera de detectar más de una pulsación de tecla, es importante conocer cómo funciona el ordenador cuya tecla está siendo pulsada. Hay dos métodos muy extendidos entre los micros domésticos. En el primer método (*polling*) el ordenador explora secuencialmente el teclado, recorriendo todas las teclas a intervalos regulares de tiempo para ver si se ha pulsado alguna.

En el segundo método la pulsación de tecla origina un mensaje llamado «interrupción» que se envía al procesador. La «interrrupción» hace que el ordenador deje por un tiempo lo que estaba haciendo para atender al teclado. Su tarea inmediata ahora es averiguar qué tecla ha sido pulsada. Este método es mucho más eficiente y versátil que el de la exploración secuencial, ya que si las teclas están inactivas no hay necesidad de prestarles atención, pero si se ha pulsado alguna tecla se le da la máxima prioridad.

Sea cual sea el método para detectar las pulsaciones de tecla, el ordenador debe poder identificar cualquier tecla con rapidez y facilidad. Casi todos los teclados utilizan un sistema llamado de generación matricial, en virtud del cual cada tecla genera un número que corresponde a su posición dentro de la matriz. Lo que sucede cuando se genera este número depende del diseño del ordenador.

Para ver qué números de la matriz generan las diferentes pulsaciones de teclas, teclea el siguiente programa:

```
20 PRINT"[SHIFT+CLR/HOME]EL
   NUMERO DE LA MATRIZ ES ";
   PEEK(197)
30 GOTO 20
```

Haz correr el programa y observa que por cada tecla que pulses se genera un número diferente. Sin embargo estos números no se ven modificados aunque pulses una tecla al mismo tiempo que SHIFT o CTRL.

El número de la tecla se almacena en la dirección 197 (como puedes ver en la línea 20), y a continuación se traduce, utilizando una tabla que hay almacenada en memoria a su correspondiente código ASCII. Este código se coloca en el *buffer* del teclado para que el procesador actúe sobre él.

Las teclas normales pueden tener significados diferentes si se pulsan al mismo tiempo que las teclas de control, tal como SHIFT. Por ello se necesitan otras tablas de traducción para decodificar estas pulsaciones dobles.

Al pulsar SHIFT, CTRL o COMM queda activado un indicador



de forma que se tenga acceso a la tabla adecuada. Sin embargo no hay una tabla de conversión que permita decodificar el número cargado en la dirección de memoria 197 cuando se pulsan dos teclas normales al mismo tiempo.

Añádele la siguiente línea al programa anterior para ver el *status* del indicador de SHIFT, que está almacenado en la dirección 653:

```
25 PRINT "VALOR DEL FLAG DE
   LA TECLA SHIFT ";
   PEEK(653)
```

Para superar esta limitación de no poder detectar más que determinadas pulsaciones simultáneas, puedes restringir el número de códigos de teclas que pueden almacenarse en el *buffer* del teclado. En condiciones normales se pueden almacenar hasta 10 caracteres, pero hay casos en que un número menor de teclas en el *buffer* permitirá al sistema operativo ignorar algunas pulsaciones y recordar únicamente las más recientes.

Puedes reducir la capacidad del *buffer* POKEando la dirección de memoria 649, que normalmente tiene un valor de 10, con un número más pequeño. Sin embargo, ten cuidado de no poner números más grandes, ya que podrías invadir las rutinas de manejo del teclado y la pantalla cercanas a esta dirección.

También podría resultar ventajoso, no ya limitarse a reducir simplemente la capacidad del *buffer*, sino vaciarlo antes de detectar una tecla. Puedes hacer esto modificando el bucle para leer un carácter del teclado:

```
10 GET A$:IF A$="" THEN 10
```

Esta línea puede resultar muy útil en un programa que dependa de la precisión de las entradas ejecutadas por el usuario.

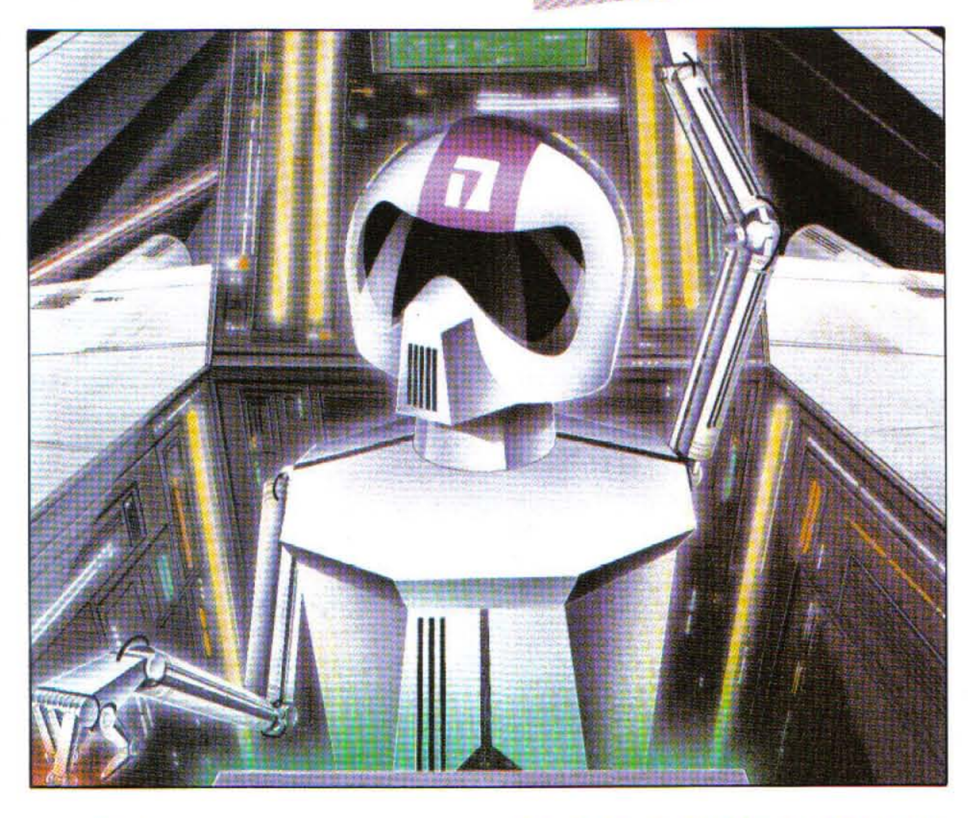
Además de vaciar el *buffer*, encontrarás de utilidad el hacer que las teclas tengan autorrepetición. Teclea POKE 650,128 como comando directo y seguidamente mantén pulsada una cualquiera de las teclas normales. Verás cómo dicho carácter se imprime repetidamente en la pantalla. Imagina que este carácter es detectado y da lugar, por ejemplo, a un rayo láser. La capacidad de autorrepetición hará que salga, no una ráfaga, sino toda una cortina de fuego de rayos laser, haciendo que el ataque sea de lo más emocionante, aunque sólo se pulse una sola tecla. Para inhibir la función de autorrepetición, pulsa POKE 650,0.

La única forma en que puedes simular las pulsaciones simultáneas de teclas en los **Commodore** es que estés preparado para utilizar las teclas SHIFT, COMM y CTRL junto con una tecla normal, lo que te da un máximo de cuatro teclas que pueden ser detectadas al mismo tiempo. Teclea el siguiente programa para ver el grado de éxito que consigues:



para Commodore-64

```
10 POKE 650,128:PRINT"[SHIFT
   +CLR/HOME]";CHR$(8)
100 P=1024:C=55296:SP=40:K1=
   37:K2=36:AX=40:AY=25:Y=
   12:X=20:POKE 53281,0
110 XX=0:YY=0:A=PEEK(197):IF
   A=K1 THEN YY=-1
120 IF A=K2 THEN YY=1
130 IF PEEK(653)=2 THEN
   XX=-1
140 IF PEEK(653)=1 THEN
```



```
    XX=1
150 IF PEEK(653)>3 THEN
    GOSUB 300
160 POKE P+(Y*SP)+X,46:POKE
    C+(Y*SP)+X,RND(1)*6+1
200 IF X+XX>-1 AND X+XX<AX
    THEN X=X+XX
210 IF Y+YY>-1 AND Y+YY<AY
    THEN Y=Y+YY
220 POKE P+(Y*SP)+X,87:GOTO
    110
300 VV=54296:WW=VVV-20:AA=WW
    +1:HH=WW-3
310 POKE VV,15:POKE WW,33:
    POKE AA,180
320 FOR Z=1 TO 15 STEP .3:
    POKE 53280,Z:POKE HH,34+
    Z:NEXT Z
330 POKE HH,0:POKE WW,0:
    RETURN
```

Teclea para Vic-20

```
100 P=7680:C=38400:SP=22:K1=
    44:K2=36:AX=22:AY=23:Y=
    10:X=11:POKE 36879,14
300 POKE 36878,15
310 FOR Z=8 TO 15 STEP .1:
    POKE 36879,Z:POKE 36877,
    200-(Z*5):NEXT Z:RETURN
```

Hay que escoger cuidadosamente las teclas a utilizar para los movimientos y para abrir fuego, ya que no se pueden detectar las pulsaciones simultáneas de teclas normales. Como nunca vas a tener que mover hacia arriba y hacia abajo al mismo tiempo, utilizaremos K y M para estas acciones. Para el movimiento a izquierda y derecha se utilizan respectivamente COMM y SHIFT, mientras que para abrir fuego se usa CTRL.

El método utilizado para detectar las cinco teclas, no responde a pulsaciones estrictamente simultáneas, pero será lo suficientemente preciso como para permitir una simulación satisfactoria. Por ejemplo, si pulsas las teclas para mover el blanco hacia arriba y hacia la izquierda, éste se desplazará en diagonal, y puedes iniciar el fuego al mismo tiempo.

GANADORES DE LOS MEJORES DE INPUT COMMODORE

En el sorteo correspondiente al número 7 realizado entre quienes escribisteis mandando vuestros votos a LOS MEJORES DE INPUT han resultado ganadores:

NOMBRE	LOCALIDAD	JUEGO ELEGIDO
Antonio Hernández Nebreda	Madrid	Reus
Pablo Salom Sánchez	Palma de Mallorca	Rambo
Manuel García Caro	Valls (Tarragona)	Rambo
Luis Astolfi Antona	Ciudad Pegaso (Madrid)	Reus
Alfredo Canteli Suárez	Oviedo	Fight Night
Jordi Boix Llinás	Gerona	Kung Fu Master
Miquel Serrá Artigas	Barcelona	Misión Imposible
Teresa Lorda De los Ríos	Bilbao	Time Tunnel
Pedro Ballesteros Madueño	Marbella (Málaga)	The Goonies
Andrés Mayoral Maestre	Madrid	Gi - Joe



Saludos, Commodorianos:

Aclarando lo del CLUB DE SOFTWARE epetimos.

Jentajas:
-Acceso al mejor y mas actual soft por una modica cantidad(500).
-Descuento de un 10 % en cualquier producto.
-Informacion periodica de todas las actividades y entrada de novedades.
-Sin limite de tiempo ni de cambios.
-Prioridad de acceso a cualquier actividad de la tienda.
-Participacion en la futura revista.
-Crear tu VIDEOTECA particular

ondiciones:
-Ingreso,compra de un programa y 500 de inscripcion.
-Todos los meses en vez de cuota compra de un programa.

PROGRAMAS

Hay un gran esprecto dividido en varios grupos:
-SIMULADORES:Revs,Elite,Jet,Silent Service,Super Huey,Flight Simulator II.....
-ESTRATEGIA:Battle of Britain,Tsushima,Broad Sides,Operation Whirl..........
-ARCADE:Con toda su amplia gama;Zeta Uridium,Yabba,Blade Runner,V..........
-AVENTURAS:Evil Crown,Enigma,Isla Te. Lord of Midnight,7 Cities,Conan..........
-CONVERSA.:Ultima III,Alternate eality Borrowed Time,Detective,Psi5,Trilog....

Y muchos mas,continuamente estan llegando importaciones.Estando disponibles,tanto en disco como en cinta.

Y sin olvidar a los usuarios del 128 y del C16,que a pesar de no tener una gran lista algo va saliendo.

HARDWARE

Por fin puedo ofreceros,despues de salvar distintos avatares,la amplia gama de productos COMMODORE,no disponibles en nuestro pais y a un precio increible.
28D-Compacto formado por un 128 y la unidad de disco 1571
570 y 1571-Las dos unidades de disco que han salido para el 128
001-Unidad de disco con 1 Mgbyte de capacidad.
572-Compacto formado por 2 1571.
902-Monitor color de Cbm dual con salida RGB y formato 40/80 clmnas.
odo con un 10% menos del precio oficial de Microelectronica.
bm64,1541,Impresora 100.000.
bm64,Cassette,Joystick 55.000

PERIFERICOS

Hay una amplia gama de perifericos para utilizar con el ordenador.

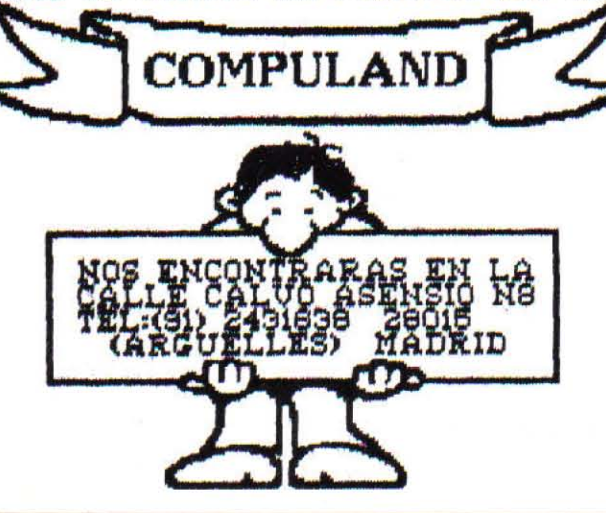
Empezando con un increible LAPIZ OPTICO,el mejor lapiz que ha pasado por mis manos,no solamente por la precision y robustez del mismo,sino por los dos maravillosos programas que lo acompanan.Programas que te permiten trabajar pixel a pixel,con opcion zoom,volcado impresora,composicion de musica y otras opciones.

Tambien estan disponibles RATONES, TABLETAS GRAFICAS,JOYSTICKS,PLOTTER TALADROS DE DISCO,CABLES IMPRESORA ALARGADORES,TAPAS y un sinfin de productos.

MUSICA

No hemos olvidado a los adicto musicales,para gran solaz y regocijo de ellos tenemos una serie de programas y sampler que hacen del crear musica un pasatiempo agradable y sencillo.

SOUND SAMPLER:Es un muestreador analogico digital,que permite almacenar cualquier sonido real(voz instrumentos-via microfono o linea) teniendo varias posibilidades como:
-asignar tonalidad,reversa,reflect...
Se puede disparar dicha informacion desde el teclado del 64,via MIDI,o desde un teclado propio.Incorpora su sistema de enlace,un secuenciador, compresor-limitador,y retardos,ecos...
SOUND EXPANDER:Amplia a 8 voces,sintesis de sonido en F.M.compone en T.R



UN CARTUCHO CON MUCHO JUEGO

- MULTIPLICADOR DE VELOCIDAD
- BOTON DE RESET
- COMANDOS DE AYUDA
- COMANDOS PARA DISCO
- INTERFACE CENTRONICS

El Commodore 64 es indudablemente uno de los microordenadores que más recursos le ofrecen al programador doméstico. Sin embargo existen algunas lagunas de mayor o menor entidad, según las necesidades del usuario.

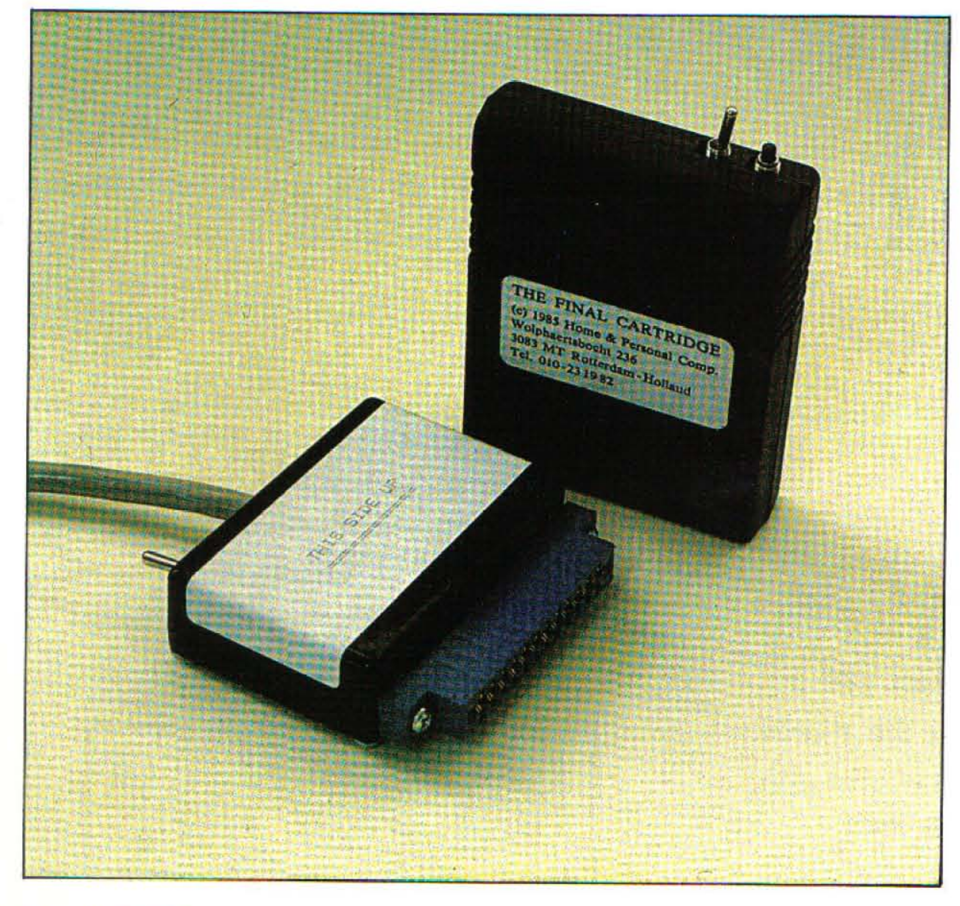
Afortunadamente, existen soluciones a nivel *soft* y *hard*. El **Final Cartridge** (Cartucho defintivo) aporta una solución mixta para ampliar las posibilidades del ordenador.

Los multiplicadores de velocidad, forman parte sin duda alguna del *hard* más apreciado y útil. Todos conocéis las características de la unidad de *diskettes*. Para solventar sus carencias han surgido en el mercado muchos «inventos», algunos empleándose a fondo en el *hardware*, otros en el *software*. Lo cierto es que *software* plantea problemas de permanencia en la memoria central, dificultades que el *hardware* elude con facilidad.

El **Final Cartridge** es un cartucho multiplicador con muchas más posibilidades. Esa es su ventaja, aunque para algunos programadores también puede ser un inconveniente.

Para empezar, las teclas de función vienen programadas igual que en el **Commodore 128**. F1 corresponde al comando LIST, F2 al monitor de lenguaje máquina, F3 a RUN, F4 es OFF, F5 proporciona directamente LOAD, F6 lo hace con SAVE, F7 muestra el directorio (DIR) y F8 es DISK.

El cartucho incorpora el útil botón de *reset*, que permite tomar el control del ordenador cuando éste se queda «colgado». No obstante, es también posible desconectarlo por *software* desde el comando OFF (tecla F4).



El cartucho incluye una serie de comandos de ayuda, pertenecientes al BASIC versión **4.0** de **Commodore**, como son AUTO (númeración automática de las líneas de programa), DEL (borra las líneas contenidas entre dos números indicados), RENUM (renumera las líneas del programa), FIND (busca una cadena o comando y lista el total de líneas en las que aparece) y HELP (detecta los errores de sintaxis).

Asimismo, el cartucho incorpora una serie de comandos para el disco, como son: CATALOG (directorio), APPEND (función de mezcla tipo *merge*), DISK (función similar a la «@» del DOS); DLOAD, DSAVE y DVERIFY (obvias).

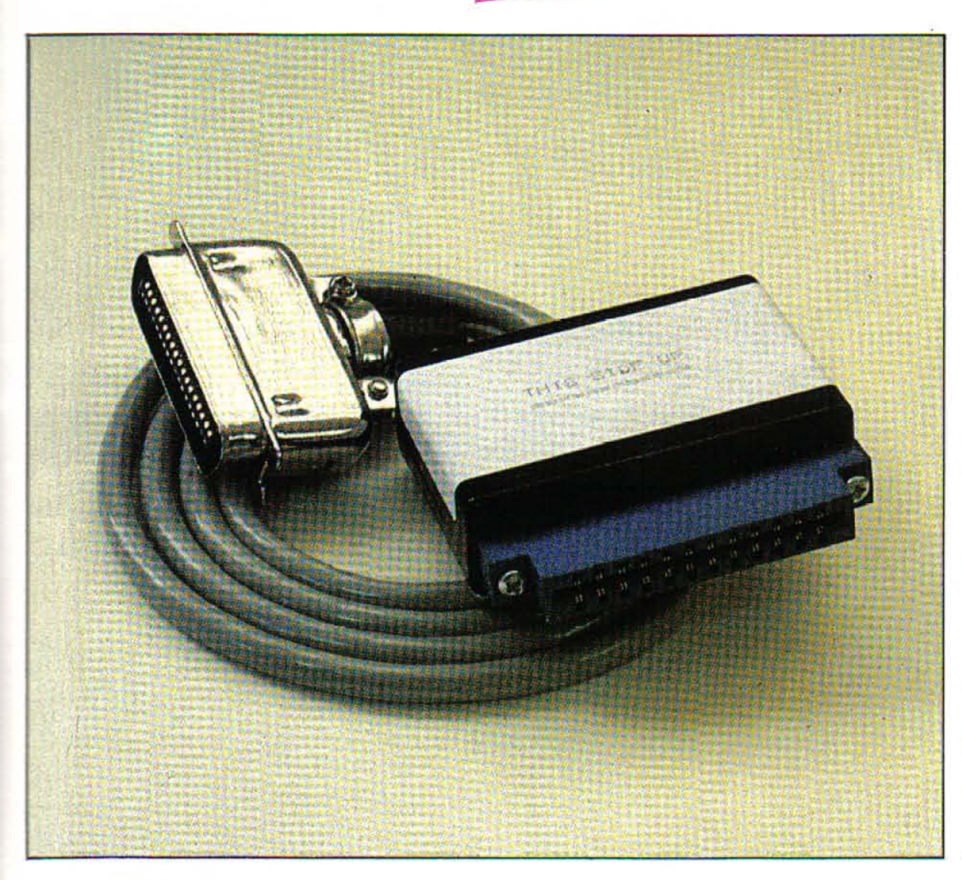
La cinta no ha sido descuidada, y el **Final Cartridge** incluye un turbo para cinta 10 veces más rápido que la velocidad normal del periférico número 1, la unidad de *cassette* **C2N**.

Este magnífico accesorio trabaja con los comandos normales: LOAD, SAVE, GET, PRINT#, etc...

El cartucho, por si fuera poco, aporta un *interface* **Centronics** y el cable adecuado para establecer la conexión con los periféricos. Este *interface* permitirá aprovechar otra opción del cartucho: el volcado de impresora, o *hardcopy*, de gráficos en alta resolución. Hay que mencionar a ese respecto que el volcado es un tanto engorroso y complicado frente a otros programas que ofrecen esta posibilidad de manera simple, pulsando una sola tecla.

Este cartucho incorpora asimismo un monitor de código máquina, muy típico en multiplicadores. utilizando los comandos habituales.

Por último, una faceta realmente



importante, la velocidad se ve incrementada por un factor de 5, aunque bien es cierto que en el comercio americano se encuentran multiplicadores como por ejemplo el **Prologic DOS**, que alcanza las 27 veces.

En efecto, el usuario de una impresora **noCommodore** dispondrá seguramente de un *interface* adecuado para manejarla. El programador de código máquina tendrá su monitor de tocado, retocado y situado en varias áreas de memoria. Los forofos de disco, o por lo menos los que conocemos, habrán comprado la **1541** para olvidarse de los innumerables problemas planteados por el *cassette*, ... etc. Con esto es de significar que un cartucho que incorpora tantos extras siempre tiene algo más que ofrecer.

Tal vez la casa holandesa que ha diseñado el cartucho debiera haber hecho más hincapié en aumentar el factor de multiplicación de velocidad de transferencia de los datos entre la unidad de *diskettes* y el ordenador.


Quick

TRAZADOR PARA EL COMMODORE

Cuando veas que tus programas se te resisten y los mensajes de error vuelan por todas partes, conecta este programa trazador y vuelve sobre tus pasos antes de naufragar en un océano de sintaxis.

Es casi imposible teclear un programa largo, sin introducir algunos errores. Por muchas comprobaciones que se hagan, siempre hay errores que desafían incluso al más hábil programador, si no cuenta con alguna poderosa herramienta de diagnóstico.

Hemos preparado un programa trazador para los poseedores del **Commodore 64** o el **Vic 20**, que os ayudará a comprobar.

El programa trazador que se lista más adelante está escrito en lenguaje ensamblador y en código máquina. Si tu ensamblador no funciona bien, puedes teclear el programa trazador en código máquina con un monitor de lenguaje máquina. Si tu ensamblador funciona bien, puedes ensamblar la traza y almacenarla de forma que puedas utilizarla para diagnosticar problemas en otros programas de BASIC que hayas escrito. Si no estás muy seguro de si tu ensamblador funciona o no, puedes probarlo intentando ensamblar el programa trazador.

COMO UTILIZARLO

Cuando tecleas RUN para ejecutar un programa en BASIC y el programa no funciona, lo más probable es que el ordenador te envíe un mensaje de error diciéndote qué línea es la que no puede ejecutar. Cuando se trata de un programa sencillo y corto, puede que esto sea todo lo que te haga falta para depurarlo completamente. Pero a medida que tus programas van siendo más largos y complicados, puede que este mensaje no te resulte suficiente para salir de las tinieblas. Puede que una línea particular sea ejecutada varias veces mientras el programa está corriendo. Y otras líneas que ya han sido ejecutadas antes, pueden asignar a las variables valores que causen problemas en la línea en la cual se para el programa.

Lo que hace el programa trazador es presentar en pantalla el número de cada línea a medida que va siendo ejecutada.

Necesitas disponer de una versión del listado de programa, ya sea sacado con una impresora o la que publicamos en **INPUT**. Utilizando la traza, sigue el programa hasta el punto en que se detiene. De esta forma podrás

- POR QUE NECESITAS UNA TRAZA
- LO QUE PUEDE HACER UNA TRAZA
- SEGUIMIENTO DE ERRORES
- FORMA DE UTILIZARLA



ver con claridad la estructura del programa. Podrás comprobar si el ordenador regresa adecuadamente de las subrutinas. También podrás averiguar los valores que toman las diversas variables durante la prueba, comprobar que las sentencias condicionales IF ... THEN se cumplen realmente y que las sentencias GOTO envían el control a la línea adecuada.

COMO FUNCIONA

Un programa trazador es algo especial. Se ejecuta mientras se está ejecutando otro programa, el programa BASIC que quieras comprobar. Normalmente no es posible ejecutar dos programas en el ordenador al mismo tiempo. Aunque parece que ambos programas están corriendo simultáneamente, en realidad no sucede así. La ejecución de la traza paraliza el programa principal utilizando lo que se llama rutinas activadas por interrupciones.

Con estas rutinas se interrumpe el programa principal cada sesentaavo de segundo. Mientras está detenido el programa principal durante una fracción de segundo se ejecuta la rutina activada por la interrupción. Cuando termina dicha rutina, continúa ejecutándose el programa principal hasta que llega la siguiente interrupción.

Los programas en BASIC siempre se interrumpen durante su ejecución. El ordenador genera una interrupción cada sesentaavo de segundo para explorar el teclado y comprobar si se ha pulsado alguna tecla. Las rutinas activadas por interrupción simplemente se unen a estas rutinas de exploración del teclado.

Debido a la alta frecuencia con que se producen las interrupciones, una línea larga de BASIC puede ser interrumpida varias veces durante su ejecución. Por ello puede suceder que el trazador te dé un número de línea repetido varias veces. Recíprocamente, si una línea es muy corta, por ejemplo un solo PRINT o un RETURN de subrutina, hay algunas probabilidades de que la traza se pierda. Algunas veces estas líneas tardan menos de un sesentaavo de segundo y podría ser de que la traza no llegara a capturarlas. Si observas que la traza no lista el número de una línea muy corta, añádele un retardo a dicha línea: un bucle FOR ... NEXT o una sentencia REM.

El siguiente programa para **Commodore 64** presenta en la esquina superior izquierda de la pantalla el número de la línea de BASIC que está siendo ejecutada en cada momento. De esta forma puedes seguir la pista de todo lo que tu programa en BASIC pueda imprimir en la pantalla.

Si tecleas el programa utilizando tu ensamblador, el origen es 49152. Si utilizas el monitor en código máquina, la dirección de partida es también 49152. Para activar la rutina, tienes que pulsar SYS 49152.

Al ejecutar el programa, los números de las líneas de BASIC se van cambiando muy rápidamente. Para detenerlos hasta una velocidad que resulte legible, pulsa la tecla [F1]. Para detenerlos, pulsa [RUN/STOP]. Si tecleas CONT el ordenador empezará otra vez en el punto donde había quedado detenido.

Teclea para Commodore-64

```
ORG 49152; $C000
        SEI                     78
        LDA #$0D             A9 0D
        STA $0314         8D 14 03
        LDA #$C0             A9 C0
        STA $0315         8D 15 03
        CLI                     58
        RTS                     60
        LDA #$01             A9 01
        LDX $D021         AE 21 D0
        CPX #$F1             E0 F1
```

```
     BNE COL              D0 02
     LDA #$00             A9 00
COL  STA $FF              85 FF
     LDA #$00             A9 00
     STA $FE              85 FE
     LDA $39              A5 39
     STA $FB              85 FB
     LDA $3A              A5 3A
     STA $FC              85 FC
     LDY #$07             A0 07
DIG  LDX #$30             A2 30
FIG  SEC                     38
     LDA $FB              A5 FB
     SBC NUMS-1,Y      F9 75 C0
     PHA                     48
     DEY                     88
     LDA $FC              A5 FC
     SBC NUMS+1,Y      F9 77 C0
     BCC SAL              90 09
     STA $FC              85 FC
     PLA                     68
     STA $FB              85 FB
     INX                     E8
     INY                     C8
     BNE FIG              D0 E8
SAL  PLA                     68
     TXA                     8A
     STY $FD        84 FD
     INC $FE        E6 FE
     LDY $FE        A4 FE
     STA $0420,Y  99 20
                     04
     LDA $FF           A5 FF
     STA $D820,Y  99 20 D8
     LDY $FD           A4 FD
     DEY                  88
     BPL DIG        10 D1
     LDA $FB        A5 FB
     ORA #$30       09 30
     STA $0425      8D 25 04
     LDA $FF           A5 FF
     STA $D825      8D 25 D8
     LDA $C5           A5 C5
     CMP #$04          C9 04
     BNE FIN           D0 0A
     LDX #$00             A2 00
OTRV LDY #$00             A0 00
ATRA INY                     C8
     BNE ATRA             D0 FD
     INX                     E8
     BNE OTRV             D0 F8
FIN  JMP $EA31         4C 31 EA
NUMS WOR 10               0A 00
     WOR 100              64 00
     WOR 1000             E8 03
     WOR 10000            10 27
```

FUNCIONAMIENTO DEL PROGRAMA

Las primeras siete instrucciones forman una pequeña rutina que pone en marcha el programa principal. La instrucción SEI activa el indicador de interrupción en el registro de *status*. De esta forma se inhiben las interrupciones normales de forma que puedas cambiar el vector de interrupción y dirigirlo hacia la rutina de trazado. Si no inhibes las interrupciones antes de hacer esto, existe el peligro de que se presente una interrupción durante el cambio y el vector apunte a un lugar que no sea el deseado.

LDA #$0D y STA $0314, y LDA #$C0 y STA $0315 sirven para cargar el número C00D en el vector de interrupción, que ocupa las direcciones de memoria 0314 y 0315 en hexadecimal. El signo # informa al ensamblador de que lo que viene detrás es un número y no una dirección, mientras que el signo $ indica que dicho número está expresado en hexadecimal. No hay una instrucción para cargar un número directamente en una dirección de memoria. Primero hay que cargar dicho número en uno de los registros A, X o Y con una instrucción LDA, LDX o LDY y a continuación se almacena el contenido del registro en la dirección de memoria adecuada con una de las instrucciones STA, STX o STY.

C00D es la dirección de comienzo de la rutina principal de la traza que empieza en los listados de lenguaje ensamblador con la instrucción LDA #$01.



Seguidamente CLI borra el indicador de interrupción, o lo que es lo mismo permite que haya interrupciones de nuevo. Con RTS se vuelve otra vez al BASIC.

Una vez que han sido ejecutadas estas siete instrucciones, queda activada la rutina de trazado. A partir de este momento cada sesentaavo de segundo, cuando el ordenador interrumpe su operación principal para explorar el teclado, se ve dirigido en cambio a la rutina de trazado.



LA RUTINA PRINCIPAL

Como puedes observar, la rutina principal no empieza inhibiendo las interrupciones como tal vez se podría esperar. Después de todo no querrás que tu rutina se vea interrumpida, ya que podría terminar en un bucle interminable. Pero en el microprocesador **6510** (y en el **6502**) las interrupciones quedan automáticamente inhibidas cuando se arranca una rutina de interrupción.

Lo primero que hace la rutina principal es comprobar el color del fondo de la pantalla: no tiene sentido escribir números blancos sobre fondo blanco; no podrías verlos.

LDA #$01 carga el registro A con el número 1, que es el número que corresponde al blanco. A continuación LDX $D021 carga el registro X con el contenido de la dirección de memoria D021 en hexadecimal, que es 53281 en decimal. Esta es la dirección que contiene el color del fondo de la pantalla. Los cuatro bits más significativos se mantienen siempre en alto, es decir a 1. De esta forma, si el color del fondo de pantalla es el blanco, el contenido de dicha dirección de memoria sería 11110001 en binario, 241 en decimal o [F1] en hexadecimal. La instrucción CPX compara el contenido del registro X, es decir el contenido de D021, con [F1].

Si el color del fondo de pantalla es el blanco, la instrucción CPX activa el indicador de cero en el registro de *status*. En caso de que el indicador de cero no haya sido activado, la instrucción BNE (*Branch if Not Equal*) salta por encima de la instrucción LDA #$00 hasta la siguiente mención de la etiqueta COL. Pero si el color del fondo es blanco y el indicador de cero ha sido activado, no se ejecuta el salto. LDA #$00 carga entonces el acumulador con 00, que es el número que corresponde al negro.

Cualquiera que sea el resultado de esta prueba, el contenido del acumulador es almacenado en la dirección de memoria FF de la página cero. El microprocesador hace referencia al número almacenado en esta dirección cuando pone un número de línea en la pantalla.

LDA #$00 y STA $FE ponen a cero el contenido de la dirección de memoria FF de la página cero. Esta dirección se utiliza como contador cuan-

para esta Primavera

ZS-48/066 (D)

ZOIDS

LA BATALLA COMIENZA

SPECTRUM 48 K

Spectrum Commodore MSX

ZS-48/090 (D)

GLADIATOR

SPECTRUM 48 K

Spectrum
Pronto: Commodore Amstrad MSX

ZS-48/085

Roller Coaster

SPECTRUM 48 K

Spectrum
Pronto: Commodore Amstrad

ZS-48/103 (D)

BOMB JACK

Sp 48K

elite

SPECTRUM 48 K

Spectrum
Pronto: Commodore Amstrad



ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141. 28046 Madrid.
Tel. 459 30 04. Tel. Barna. 209 33 65. Telex: 22690 ZAFIR E

Programas editados, fabricados y distribuidos en España con la garantía Zafiro.

do la rutina calcula la posición de los dígitos decimales sobre la pantalla.

LDA $39 y STA $FB, y LDA $3A y STA $FC transfieren el contenido de las direcciones de memoria 39 y 3A en hexadecimal a FC y FB. Las direcciones de memoria 57 y 58 contienen el número de la línea de BASIC que el ordenador está ejecutando en cada momento. FC y FB son direcciones libres de la página cero, en la que pueden ser manipulados estos números; recuerda que estos números están en hexadecimal y que la rutina tiene que hacer una considerable cantidad de trabajo sobre ellos para presentarlos en pantalla en forma decimal. Se supone que no querrás que el programa trazador te presente los números de las líneas en forma hexadecimal.

La siguiente sección calcula los números decimales que serán pokeados en la pantalla.

LDY #$07 carga el registro Y con el número 7. Este registro se utiliza como valor de desplazamiento y como contador, de forma que la rutina sepa en cada momento con qué digito decimal está trabajando. LDX #$30 almacena el número decimal 48 en el registro X. 48 es el código ASCII del número 0 y nuevamente se utiliza el registro X como contador, recorriendo en su cuenta los valores del código ASCII para que se ponga el dígito correcto en esa posición de la pantalla.

La instrucción SEC (*Set Carry*) activa el indicador de *carry* (acarreo). Siempre hay que hacer esto antes de efectuar una resta, ya que es precisamente el 1 del indicador de acarreo el que se suma a los bits invertidos para obtener el complemento a 2. Cuando el microprocesador tiene que hacer una resta, lo que hace es cambiar lo ceros por unos y viceversa, y después hace una suma. Dicha suma tiene en cuenta el acarreo, por ello si este indicador está puesto a 1, dicho 1 se suma realmente para dar el complemento a 2. Cuando el indicador no está activado, es decir cuando es 0, lo que se suma es 0, lo cual es equivalente a restar un 1 de más.

Todo lo anterior resulta muy útil cuando se restan números de dos bytes. Si pones a 1 el acarreo, puedes restar primero los bytes bajos y después los bytes altos. Si ahora en los bytes altos necesitas «llevarte» (*borrow*) 1, el indicador de acarreo deberá ponerse a cero. Cuando se restan los bytes altos, se restará un 1 de más y «la que te llevabas» será tenida en cuenta automáticamente. Observa que el indicador de acarreo funciona de manera opuesta a la que lógicamente podría esperarse. Vale 1 cuando no te llevas nada, y vale 0 cuando te llevas 1 en la resta.

La regla práctica es pues la siguiente: si vas a restar, activa el indicador de acarreo, poniéndolo a 1. Si vas a sumar, ponlo a 0, ya que no querrás sumar un 1 de más.

La siguiente cosa que ocurre en esta rutina es exactamente una resta de dos bytes del tipo que acabamos de describir. LDA $FB carga el acumulador con el contenido de FB que corresponde al byte bajo del número de línea en curso. SBC NUMS-1,Y resta el byte bajo de un número que extrae de la tabla que comienza en la etiqueta NUMS. Dicha tabla contiene los números WOR al final de la rutina. Cada uno de los números decimales que aquí se dan, se carga en dos bytes en forma hexadecimal. La instrucción SBC mira el bit que quieres restar, saltando al byte anterior a la etiqueta NUMS, es decir a la dirección de memoria NUMS-1 y contando desde allí un número de direcciones igual al contenido en el registro Y. Así, en el primer paso, cuando el contenido de Y es 7, resta el byte seis (7−1) de NUMS, que es el byte bajo de WOR 10000. Puede que te parezca algo extraño utilizar estos dos desplazamientos, el -1 con la etiqueta y el contenido del registro Y, pero así se ahorran instrucciones y se obtiene el valor correcto del registro Y al final de esta rutina. La utilizamos para memorizar cuál es el dígito decimal sobre el que la rutina está trabajando.

El resultado de la primera resta se transfiere al *stack* (pila) con la instrucción PHA. A continuación se decrementa el contenido del registro Y con DEY. Finalmente LDA $FC y SBC NUMS+1,Y restan el byte alto del mismo decimal de la tabla.

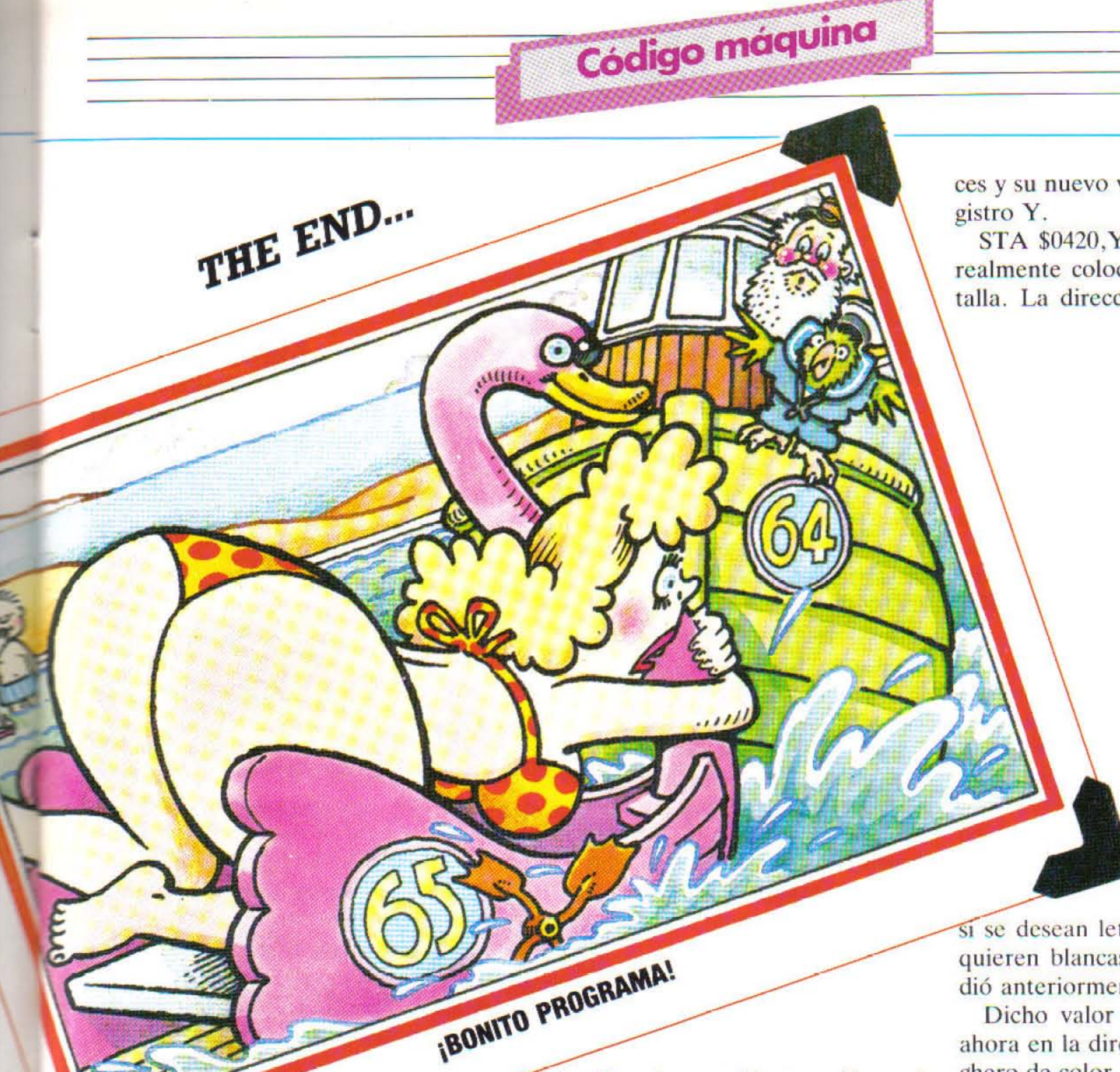
La instrucción BCC significa salto si el acarreo es cero (*Branch on Carry Clear*). Según esto si el acarreo es 1 la rutina salta al siguiente punto donde se mencione la etiqueta OUT que aparece antes de PLA. Recuerda bien que el indicador de acarreo es 0 (o *clear*) cuando te llevas 1. Si no te llevas nada y el indicador de acarreo vale 1, la rutina avanza hasta la siguiente instrucción.

STA $FC vuelve a poner el resto de este byte alto en FC. PLA y STA $FB saca del *stack* el resto del byte bajo y lo carga en FB. INX incrementa el registro X. El efecto conseguido es hacer que el registro X vaya contando los códigos ASCII de los dígitos decimales.

INY incrementa el registro Y. Esto se hace debido a que dicho registro fue anteriormente decrementado y hay que restaurar su valor antes de que la siguiente instrucción BNE FIG envíe al procesador nuevamente al bucle.

AHORRANDO BYTES

BNE significa salto si el resultado no es igual a cero (*Branch if Not Equal*). El resultado de incrementar el registro Y solamente puede ser igual a cero si anteriormente había sido -1. Si examinas detalladamente cómo funciona este programa, verás que esto no ocurre nunca. Utilizando un salto con direccionamiento relativo, en vez de un salto incondicional —en el cual



se usa direccionamiento absoluto— se ahorra un byte.

Este salto condicional significa que el microprocesador recorre el bucle una y otra vez, tomando los valores decimales que encuentra en la tabla de datos, incrementando el registro X y contando los códigos ASCII. Como puedes ver, la forma en que trabaja esta rutina es contar cuántas veces puede quitar 10000, cuantas veces puede quitar 1000, cuántas veces puede quitar 100 y cuántas veces puede quitar 10. Cuando hace todas estas restas repetidas y se encuentra que ya no puede seguir, la instrucción BCC le hace salir del bucle a la instrucción PLA. Esta instrucción transfiere el byte bajo desde el *stack* al registro A. Inmediatamente se transfiere el contenido del registro X al acumulador con la instrucción TXA.

La única razón por la que se saca el byte bajo del *stack* es evitar que éste se haga demasiado grande.

Cuando la resta se detiene, el registro X contiene el valor ASCII del dígito correcto cuyo valor decimal se está calculando. La instrucción TXA pone ese valor en el acumulador, dejándolo listo para darle salida.

PONIENDOLO EN PANTALLA

STY $FD almacena temporalmente el valor del contador decimal en FD de forma que el registro Y se pueda utilizar en algo más. El contador de posiciones de FE se incrementa entonces y su nuevo valor se carga en el registro Y.

STA $0420,Y es la instrucción que realmente coloca el dígito en la pantalla. La dirección de memoria 0420 está en la zona de pantalla, en la parte superior derecha de la misma. El código ASCII del dígito que la rutina ha estado calculando se almacena en las direcciones de memoria sucesivas añadiéndole el desplazamiento Y. De esta forma se pokea el dígito correcto en la posición correcta de la pantalla.

LDA $FF carga el acumulador con el contenido de FF, que será cero si se desean letras negras y uno si se quieren blancas. El color ya se decidió anteriormente.

Dicho valor de color se almacena ahora en la dirección adecuada del fichero de color. La memoria de pantalla y la memoria de color trabajan de forma totalmente independiente. Lo que se presenta en cualquier posición de la pantalla es completamente independiente del color con que aparece. Y el hecho de que el color se fije después de que el dígito ha sido pokeado en la pantalla no influye. Como toda la rutina tarda menos de un sesentaavo de segundo, no hay posibilidad de que aparezca un dígito en la pantalla y después, una fracción de segundo más tarde, cambie de color. Tus ojos no pueden responder con tanta rapidez, ni tampoco la pantalla de tu televisor.

El registro Y se carga otra vez con el contenido de FD y se decrementa de nuevo. Esta es la posición decimal almacenada que cuenta 7 hacia abajo.

BPL quiere decir salto si el resultado el positivo (*Branch on Plus result*).

considerando también el cero como positivo. Así, la instrucción BPL DIG envía al micro al principio de toda la rutina si no está sobre los dígitos segundo a último. En otras palabras, vuelve a ejecutar el bucle para hallar el siguiente número, a menos que el último dígito sea el de las decenas. Las unidades se calculan con una rutina separada que es la que sigue inmediatamente después.

LDA $C5 carga el acumulador con lo que quede después de calcular las decenas. Seguidamente se le hace un OR con 48 para obtener el valor ASCII. Se emplea una instrucción ORA en vez de ADC (*Add with Carry*, suma con acarreo), que es la que tendrías que usar para poner a cero el indicador de acarreo antes de sumar. Por eso haciendo un OR, que no se ve afectado por el indicador de acarreo, se ahorra un byte.

El resultado se almacena ahora en la posición de las unidades de la pantalla y se le añade el color correcto.

La dirección C5 de la página cero de memoria contiene detalles de la última tecla que ha sido pulsada, el número que haya aquí almacenado depende realmente de la situación de la tecla dentro del teclado y no de su correspondiente código ASCII.

Los contenidos de C5 se cargan en el acumulador y se comparan con 4, que corresponde a la tecla [F1]. BNE FIN salta sobre la rutina de retardo siguiente en caso de que no haya sido pulsada la tecla [F1] y el correspondiente resultado de CMP $04 no sea 0. La rutina de retardo contiene un bucle dentro de otro bucle. Los registros X e Y se cargan con 0 y se incrementan. El microprocesador realiza un salto condicional cada vez que el resultado del incremento es distinto de cero. Obviamente solo se tendrá un cero cuando el registro se llene y se desborde. Esto ocurre cada 256 veces que se recorre el bucle, lo cual significa que el bucle interior del registro Y se recorre en total 256×256 veces.

La dirección de memoria EA31 es el punto de partida de la rutina de interrupción regular. JMP $EA31 envía al micro a ejecutar la rutina de interrupción regular. Si no hicieras esto, el teclado se congelaría y no podrías usar el ordenador.

Lo que sigue son los datos decimales utilizados para calcular los dígitos. Los datos se introducen como palabras (WORds). WOR no es propiamente lenguaje ensamblador. Es una instrucción de ensamblador que le dice que deje dos bytes vacíos para los datos.

Habrás observado que en este programa no hay una instrucción de retorno y es posible que te intrigue la forma en que retorna al BASIC. Por otra parte habrás sospechado que la instrucción de retorno se encuentra en la rutina de interrupción regular. Una vez que ha realizado sus tareas normales —como la exploración del teclado— automáticamente habilita las interrupciones y vuelve directamente al BASIC sin volver a la rutina que tecleaste. Entonces el ordenador hace otro sesentaavo de segundo de BASIC y realiza nuevamente la rutina de interrupción.

FORMA DE UTILIZAR LA RUTINA

Para comprobar tu ensamblador o cualquier otro programa cuya traza quieras seguir, tienes que cargar primero la rutina en la memoria del ordenador. Si tu ensamblador funciona bien puedes cargar la rutina de la forma habitual. De no ser así, tendrás que teclear los números hexadecimales dados, utilizando tu monitor de código máquina.

Observa que la rutina de la traza está en la zona protegida detrás de C000, por lo que no tienes que modificar los límites del BASIC. Cuando termines de introducirla puedes salvarla y hacer un NEW para abandonar el ensamblador o el monitor. Ahora puedes meter el programa BASIC cuya traza quieres seguir.

Puedes activar la traza con SYS 49152 y ejecutar el programa. Ejecútalo con secciones de programa que sepas que funcionan adecuadamente. Pulsa la tecla F1 cuando te acerques a las partes con las que tienes dificultades. Si quieres congelar la traza, pulsa la tecla [RUN/STOP].

Cuando hagas la traza asegúrate antes de que tienes ante tí un listado del programa para poder ver lo que está sucediendo, ya que en caso contrario los números que aparecen en rápida sucesión en la esquina superior derecha de la pantalla no significarían nada para tí.

Si el programa te pide que introduzcas algo en la línea superior, utiliza en su lugar la segunda línea ya que no podrás escribir sobre los dígitos de la traza.

Para inhibir la traza pulsa [RUN/STOP] y [RESTORE].

El programa de trazado para el **Commodore 64** correrá también sobre el **Vic 20**, con unas ligeras modificaciones. Como el programa es básicamente el mismo la explicación dada para el **Commodore 64** vale también para la traza del **Vic 20**.

El primer cambio que tienes que hacer es reubicar todo el programa. En el **Vic** la dirección C000 ya no corresponde a una zona protegida. Por ello tienes que pokear 52 y 56 con 28 y usar 7168 como dirección de partida. Para ejecutar el programa tienes que llamar a SYS 7168. El vector de interrupción ha sido así cambiado a la nueva dirección de partida. El segundo LDA es ahora LDA #$1C.

Casi todos los cambios tienen que ver con la pantalla que en el **Vic** está en una situación diferente. Por eso cuando quieras modificar el color de la pantalla tendrás que cargar el registro X con el contenido de 900F, lo que haces con la instrucción LDX $900F, en vez de LDX $D021.

El problema es que en el Vic **900F** contiene el color del borde de pantalla así como el de la propia pantalla. Por eso tienes que comprobar si el valor que contiene es menor que 16, que quiere decir que el color de la tinta es el blanco. Las instrucciones CPX $F1 y BNE COL del 64 han de ser reemplazadas con:

DATA BECKER

FERRE MORET, S.A.

C/. TUSET, 8-10, ENTL. 2.

TEL. 218 02 93 - 08006 BARCELONA

Estos son nuestros BESTSELLERS

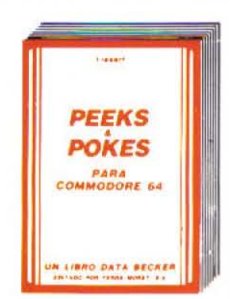












OTROS TITULOS



¿Ud. ha logrado iniciarse en código máquina? Entonces el «nuevo English» le enseñará cómo convertirse en un profesional. Naturalmente con muchos programas ejemplo, rutinas completas en código máquina e importantes consejos y trucos para la programación en lenguaje máquina y para el trabajo con el sistema operativo.
Lenguaje máquina para avanzados CBM 64. 1984, 206 pág. P.V.P. 2.200 ptas.



Este libro ofrece una amplia práctica introducción en el importante tema de la gestión de ficheros y bancos de datos, especialmente para los usuarios del Commodore 64. Con muchas interesantes rutinas y una confortable gestión de ficheros.
Todo sobre bases de datos y gestión de ficheros para Commodore-64. 221 págs. P.V.P. 2.200,- ptas.



Gráficos para el Commodore 64 es un libro para todos los que quieren hacer algo creativo con su ordenador. El contenido abarca desde los fundamentos de la programación de gráficos hasta el diseño asistido por ordenador (CAD).
Gráficos para el Commodore 64. 295 pags. P.V.P. 2.200,- ptas.



Para los usuarios que posean un VIC-20, C-64 o PC-128 este libro contiene gran cantidad de consejos, trucos, listados de programas, así como información sobre Hardware, tanto si usted dispone de una impresora de margarita o de matriz, como si tiene un Plotter VC-1520, el GRAN LIBRO DE IMPRESORAS constituye una inestimable fuente de información.
Todo sobre impresoras. 361 págs. P.V.P. 2.800,- ptas.



Este libro presenta una detallada e interesante introducción a la teoría, conceptos básicos y posibilidades de uso de la inteligencia artificial (1A). Desde un resumen histórico sobre las máquinas «pensantes» y «vivientes» hasta programas de aplicación para el Commodore 64.
Inteligencia artificial. 395 págs. 2.800,- ptas.



64, Consejos y Trucos vol. 2 contiene una gran profusión de programas, estímulos y muchas rutinas útiles. Un libro que constituye una ayuda imprescindible para todo aquél que quiera escribir programas propios con el COMMODORE.
Consejos y Trucos, Commodore 64. Vol. 2. 259 págs. 2.200,- ptas.



En el libro de los robots se muestran las asombrosas posibilidades que ofrece el CBM 64, para el control y la programación, presentadas con numerosas ilustraciones e intuitivos ejemplos. El punto principal: Cómo puede construirse uno mismo un robot sin grandes gastos. Además, un resumen del desarrollo histórico del robot y una amplia introducción a los fundamentos cibernéticos. Gobierno del motor, el modelo de simulación, interruptor de pantalla, el Port-Usuario cómodo del modelo de simulación, Sensor de infrarrojos, concepto básico de un robot, realimentación unidad cibernética, Brazo prensor, Oír y ver.
Robótica para su Commodore 64. 340 págs. P.V.P. 2.800 ptas.



Saberse apañar uno mismo, ahorra tiempo, molestias y dinero, precisamente problemas como el ajuste del floppy o reparaciones de la platina se pueden arreglar a menudo con medios sencillos. Instrucciones para eliminar la mayoría de perturbaciones, listas de piezas de recambio y una introducción a la mecánica y a la electrónica de la unidad de disco, hay también indicaciones exactas sobre herramientas y material de trabajo. Este libro hay que considerarlo en todos sus aspectos como efectivo y barato.
Mantenimiento y reparación del Floppy 1541. 325 págs. P.V.P. 2.800,- ptas.



Este es el libro que buscaba: un diccionario general de micros que contiene toda la terminología informática de la A a la Z y un diccionario técnico con traducciones de los términos ingleses de más importancia - los DICCIONARIOS DATA BECKER prácticamente son tres libros en uno. La increíble cantidad de información que contienen, no sólo los convierte en enciclopedias altamente competente, sino también en herramientas indispensables para el trabajo. El DICCIONARIO DATA BECKER se edita en versión especial para APPLE II, COMMODORE 64 e IBM PC.
El diccionario para su Commodore 64. 350 pág. P.V.P. 2.800,- ptas.



Ofrece un campo fascinante y amplio de problemáticas científicas. Para esto el libro contiene muchos listados interesantes. Análisis de Fornier y síntesis, análisis de redes, exactitud de cálculo, formateado de números, cálculo del valor PH, sistemas de ecuaciones diferenciales, modelo ladrón presa, cálculo de probabilidad, medición de tiempo, integración, etc.
64 en el campo de la Técnica y la Ciencia. 361 págs. P.V.P. 2.800,- ptas.



Este libro, contiene muchos interesantes programas de aprendizaje para solucionar problemas, descritos detalladamente y de manera fácilmente comprensible. Temas: progresiones geométricas, palanca mecánica, crecimiento exponencial, verbos irregulares, ecuaciones de segundo grado, movimientos de péndulo, formación de moléculas, aprendizaje de vocablos, cálculo de interés y su capitalización.
Manual escolar para su Commodore 64. 389 págs. P.V.P. 2.800,- ptas.



Casi todo lo que se puede hacer con el Commodore 64, está descrito detalladamente en este libro. Su lectura no es tan sólo tan apasionante como la de una novela, sino que contiene, además de listados de útiles programas, sobre todo muchas, muchas aplicaciones realizables en el C64. En parte hay listados de programas listos para ser tecleados, siempre que ha sido posible condensar «recetas» en una o dos páginas. Si hasta el momento no sabía que hacer con su Commodore 64, ¡después de leer este libro lo sabrá seguro!
El libro de ideas del Commodore 64. 1984, más de 200 páginas, P.V.P. 1.600,- ptas.

Teclea para Commodore-64

```
DIRECCION INICIAL: C000

78 A9 0D 8D 14 03 A9 C0
8D 15 03 58 60 A9 01 AE
21 D0 E0 F1 D0 02 A9 00
85 FF A9 00 85 FE A5 39
85 FB A5 3A 85 FC A0 07
A2 30 38 A5 FB F9 75 C0
48 88 A5 FC F9 77 C0 90
09 85 FC 68 85 FB E8 C8
D0 E8 68 8A 84 FD E6 FE
A4 FE 99 20 04 A5 FF 99
20 D8 A4 FD 88 10 D1 A5
FB 09 30 8D 25 04 A5 FF
8D 25 D8 A5 C5 C9 04 D0
0A A2 00 A0 00 C8 D0 FD
E8 D0 F8 4C 31 EA 0A 00
64 00 E8 03 10 27
```

Teclea para Vic-20

```
CPX #$10
BCC COL
```

Como el área de pantalla está en una posición distinta la instrucción STA $040F,Y del **Commodore 64** se convierte ahora en STA $1E0F,Y. La memoria de color también cambia de sitio, por lo que STA $D820, Y se convierte ahora en STA $960F,Y. Asimismo STA $0425 y STA $D825 pasan a ser STA $1E14 y STA $9614.

El teclado del **Vic** tiene una configuración ligeramente diferente del **C-64**, por lo que la tecla F1 no da el valor 4 sino el 39. Por ello la instrucción CMP #$04 debe ser ahora CMP #$27, La rutina de interrupción regular empieza en el **Vic** en EABF en lugar de la EA31. La instrucción JMP $EA31 pasa a ser pues JMP $EABF.

Por lo demás, el programa trazador del **Vic 20** opera de una forma exactamente igual que el del **Commodore 64**.

Naturalmente, tendrás que ensamblar a mano el trazador para el **Vic**, a menos que te hayas comprado un ensamblador comercial. A continuación te presentamos la versión del programa en código máquina a fin de que puedas comprobar tu conversión o simplemente teclees directamente la versión hexadecimal.

Teclea para Commodore-64

```
DIRECCION INICIAL:7168

78 A9 0D 8D 14 03 A9 1C
8D 15 03 58 60 A9 01 AE
0F 90 E0 10 90 02 A9 00
85 FF A9 00 85 FE A5 39
85 FB A5 3A 85 FC A0 07
A2 30 38 A5 FB F9 75 1C
48 88 A5 FC F9 77 1C 90
09 85 FC 68 85 FB E8 C8
D0 E8 68 8A 84 FD E6 FE
A4 FE 99 0F 1E A5 FF 99
0F 96 A4 FD 88 10 D1 A5
FB 09 30 8D 14 1E A5 FF
8D 14 96 A5 C5 C9 27 D0
0A A2 00 A0 00 C8 D0 FD
E8 D0 F8 4C BF EA 0A 00
64 00 E8 03 10 27
```

LA PROGRAMACION PARA JOYSTICKS

- COMO LEER LOS JOYSTICKS DESDE BASIC
- HAZ TUS JUEGOS MAS PROFESIONALES
- EL PULSADOR DE FUEGO

Los *joysticks* son la clave para obtener juegos más profesionales. Pero hacer que tus juegos resulten más divertidos no significa que tengas que aprender código máquina; puedes empezar con el BASIC.

Una diferencia evidente entre los juegos comerciales y los producidos en casa es con frecuencia la existencia de una opción para *joystick*. Sin embargo no tienes que sumergirte en las profundidades intrincadas del código máquina para utilizar los *joysticks* en tus propios programas.

En esta sección del coleccionable, veremos la manera de utilizar los *joysticks* con programas escritos en BASIC, haciendo que tus programas de juegos resulten más profesionales y más divertidos de jugar.

En el próximo capítulo utilizaremos la rutina de *joystick* en un juego, por lo que no debes de olvidarte de almacenar el programa.

Lo primero que necesitas es un *joystick* adecuado. Todo programa está escrito para adaptarse a las características de los *joysticks* indicados.

CONECTA EL JOYSTICK

El programa del **Commodore 64** es compatible con cualquier *joystick* tipo **Atari**. En primer lugar conecta tu *joystick* en el zócalo marcado con 1.

Teclea ahora la primera sección del programa que te permitirá utilizar tu *joystick* para mover un punto de mira por la pantalla.

En la segunda parte de este artículo veremos la forma de incorporar el programa en un juego completo de caza de patos.

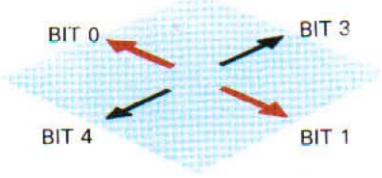
Teclea para Commodore-64

```
20 POKE 53280,5:POKE 53281,0
   :PRINT"[SHIFT+CLR/HOME]"
30 Y=1284:TU=0:SC=0
40 FOR Z=1 TO 23:PRINT"[CTRL
   +8][COMM++][38*ESPACIO]
   [COMM++]";:NEXTZ
110 J=PEEK(145)
120 POKE Y,32:POKE Y+1,32:
    POKE Y+40,32:POKE Y+41,
    32
130 T=0:T1=0:T2=0:T3=0:T4=0:
    IF ((J AND 1)=0)=-1 THEN
    T1=-40
140 IF((J AND 2)=0)=-1 THEN
    T2=40
150 IF((J AND 4)=0)=-1 THEN
    T3=-1
160 IF((J AND 8)=0)=-1 THEN
    T4=1
165 T=T1+T2+T3+T4
170 IF(Y+T>1063 AND Y+T<1943
    )AND(PEEK(Y+T)<>102 AND
    PEEK(Y+T+1)<>102)THEN
    Y=Y+T
180 POKE Y,125:POKE Y+1,109:
    POKE Y+40,110:POKE Y+41,
    112
190 GOTO 110
```

La línea 20 establece los colores de fondo y borde de pantalla y borra ésta por completo. La posición de partida del punto de mira se define en la línea 30, así como el número de patos y la puntuación; las variables TU y SC no serán necesarias hasta más adelante. La línea 40 se ocupa de dibujar el borde.

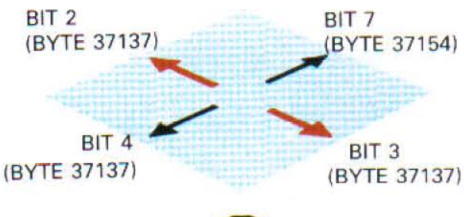
La rutina de las líneas 110 a 180 se ocupa del movimiento del *joystick*. El PEEK de la dirección 145 permite obtener la información sobre cuál de las cuatro direcciones arriba, abajo, izquierda o derecha, corresponde al movimiento del *joystick*, así como de averiguar si se ha pulsado o no el disparador.

Para detectar el movimiento del *joystick* se utiliza el operador lógico AND. El bit cero se activa cuando el *joystick* se empuja hacia arriba. Por eso al hacer AND con 1 el resultado será cero al empujar el *joystick* en esa dirección. Análogamente, para el movimiento hacia abajo se pone a 1 el bit 1 y hay que hacer la operación AND con 2. Para el movimiento hacia la izquierda se hace el AND de J con 4 y para la derecha se hace el AND de J con 8. Como veremos en la siguiente parte de nuestro coleccionable, hay que hacer un AND de J con 16 para





El joystick *del* **Commodore 64** *pone a uno estos bits en la posición 145. Al hacer AND con ellos podrás saber en qué dirección se está moviendo.*





El joystick *del* **Vic 20** *actúa sobre los bits de dos direcciones separadas de memoria, dependiendo de la dirección en que se está moviendo.*

detectar las pulsaciones del disparador.

En la línea 130 T, T1, T2, T3 y T4 se ponen a cero. Este conjunto de variables se utilizan para ajustar sobre la pantalla la posición del punto de mira, según el resultado del AND con J. Las líneas 130 y 140 se ocupan de los movimientos verticales del punto de mira, mientras que las líneas 150 y 160 son las que se ocupan del movimiento hacia la izquierda y hacia la derecha.

La línea 165 calcula el cambio global en la posición de la pantalla. Siempre que el punto de mira no vaya a salir fuera de la pantalla, la línea 170 calcula una nueva posición de pantalla. Los POKEs de la línea 180 visualizan nuevamente el punto de mira.

El programa del **Vic 20** es muy parecido al programa que hemos presentado para el **Commodore 64**. Como de costumbre, algunas de las direcciones de memoria son diferentes y las dimensiones de la pantalla también son diferentes.

PUNTO DE MIRA

Teclea el siguiente programa y tendrás un punto de mira que se mueve por la pantalla:

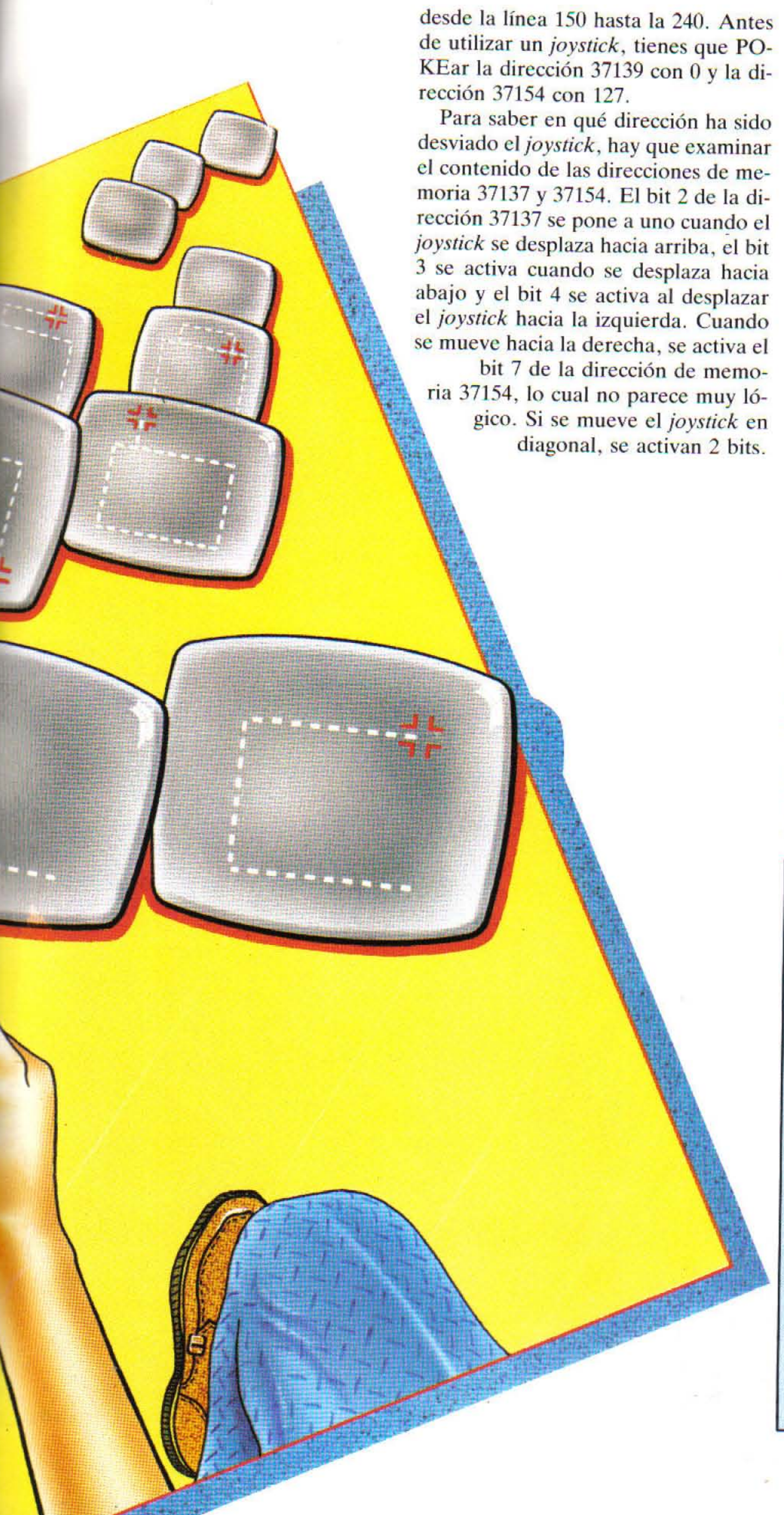
Teclea para Vic-20

```
40 FOR Z=0 TO 7:POKE 7424+Z,
   0:NEXT Z
50 POKE 36879,13:PRINT"
   [SHIFT+CLR/HOME]":POKE
   36869,255
60 Y=7791+88*2:POKE 36878,15
   :TU=0:SC=0
70 FOR Z=1 TO 22:POKE 7680+Z
   *22,230:POKE 7701+Z*22,
   230:NEXT Z
150 POKE 37139,0:POKE 37154,
    127:J=PEEK(37136):J0=(
    (J AND 4)=0:J1=((J AND
    8)=0)
160 J2=((J AND 16)=0:F=((J
    AND 32)=0):J3=((PEEK(
    37152)AND 128)=0:POKE
    37154,255
170 POKE Y,32:POKE Y+1,32:
    POKE Y+22,32:POKE Y+23,
    32
180 T=0:T1=0:T2=0:T3=0:T4=0:
    IF J0=-1 THEN T1=-22
190 IF J1=-1 THEN T2=22
200 IF J2=-1 THEN T3=-1
210 IF J3=-1 THEN T4=1
220 T=T1+T2+T3+T4
230 IF(Y+T>7701 AND Y+T<8163
    )AND(PEEK(Y+T)<>230)THEN
    Y=Y+T
240 POKE Y,253:POKE Y+1,237:
    POKE Y+22,238:POKE Y+23,
    240
250 GOTO 150
```

En primer lugar la línea 40 define el espacio de caracteres. La línea 50 establece el color de la pantalla, la deja limpia y activa el modo gráfico. La línea 60 es otra línea de inicialización, referente a la posición de partida del punto de mira, variable Y, y a otras cosas que no serán necesarias hasta el siguiente artículo: el volumen de los efectos sonoros, el número de patos, TU, y la puntuación SC. La línea 70 define el borde de pantalla.

La rutina del *joystick* se extiende





desde la línea 150 hasta la 240. Antes de utilizar un *joystick*, tienes que POKEar la dirección 37139 con 0 y la dirección 37154 con 127.

Para saber en qué dirección ha sido desviado el *joystick*, hay que examinar el contenido de las direcciones de memoria 37137 y 37154. El bit 2 de la dirección 37137 se pone a uno cuando el *joystick* se desplaza hacia arriba, el bit 3 se activa cuando se desplaza hacia abajo y el bit 4 se activa al desplazar el *joystick* hacia la izquierda. Cuando se mueve hacia la derecha, se activa el bit 7 de la dirección de memoria 37154, lo cual no parece muy lógico. Si se mueve el *joystick* en diagonal, se activan 2 bits.

En las líneas 150 y 160 se utiliza el AND lógico para detectar la forma en que se mueve el *joystick*; las variables J0 a J3 corresponden a arriba, abajo, izquierda y derecha respectivamente. Se comprueba además el disparador de fuego. Para ello se activa el bit 5 de la dirección de memoria 37137 cuando se acciona dicho disparador. Si el bit que está siendo examinado vale 1, la correspondiente variable J toma el valor −1.

En la línea 180 se ponen a cero las variables T, T1, T2, T3 y T4. Este conjunto de variables se utiliza para ajustar la posición del punto de mira sobre la pantalla, con arreglo a los valores que tomen J0, J1, J2 y J3. Las líneas 180 y 190 corresponden a los movimientos verticales del punto de mira, mientras que las líneas 200 y 210 corresponden a los desplazamientos a izquierda y derecha.

La línea 220 calcula el cambio en la posición de pantalla. Si el punto de mira no va a salir fuera de la pantalla, la línea 230 calcula una nueva posición. Los POKE de la línea 240 presentan el punto de mira en la pantalla.

La línea 250 no es más que una medida temporal para que el *joystick* pueda controlar continuamente al punto de mira.

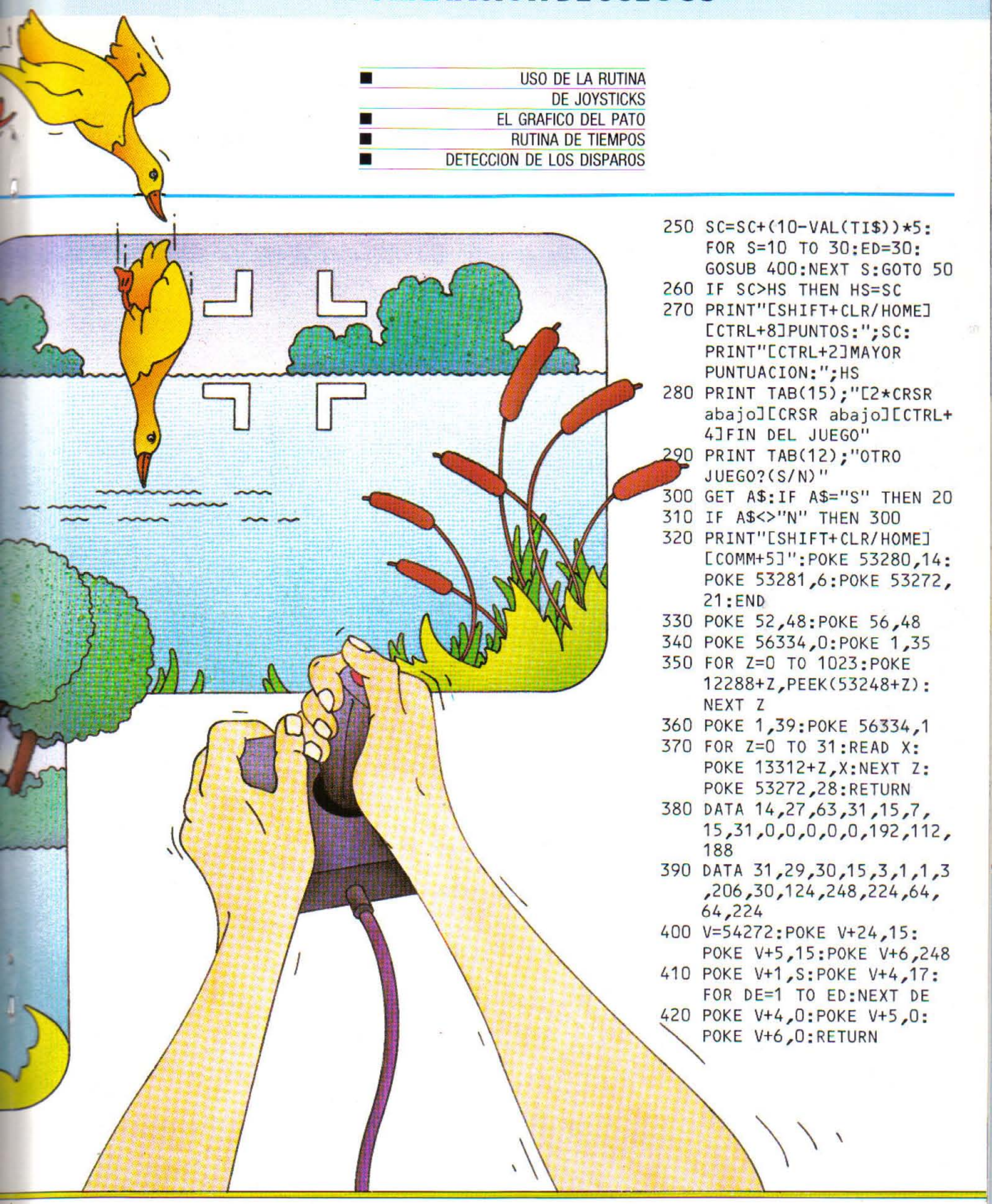
P y R

¿Se le puede agregar una rutina de *joystick* a cualquiera de los juegos que aparecen en INPUT?

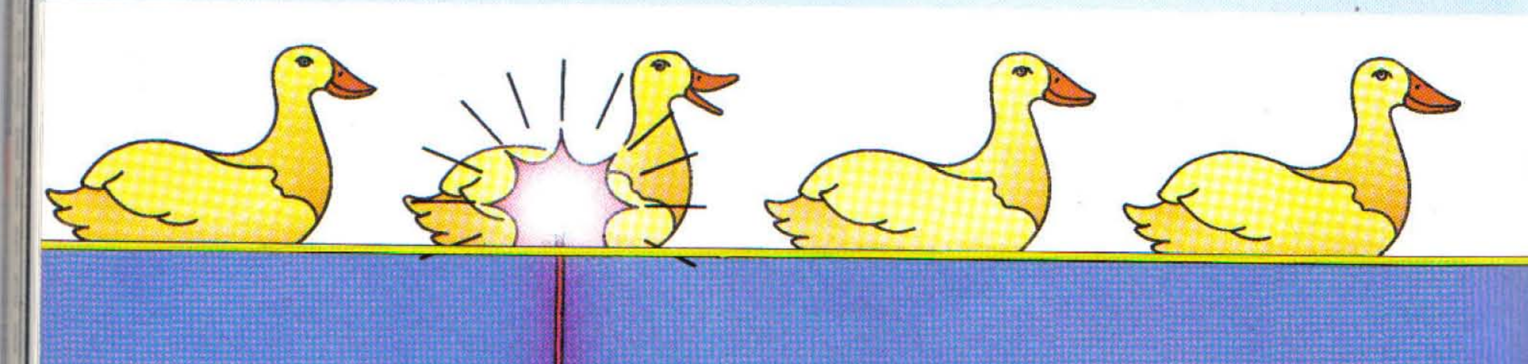
El control por *joystick* no es más que una alternativa conveniente al control por teclado, por lo que puedes sustituir el núcleo de la mayoría de los programas que aparecen en nuestro coleccionable en los cuales se utilice GET$ para leer el teclado.

Las líneas importantes son la 110 y de la 130 a la 170 para el **Commodore 64** y de la 150 a la 240 para el **Vic 20** (se utiliza la variable Y para POKEar en la pantalla).

- USO DE LA RUTINA DE JOYSTICKS
- EL GRAFICO DEL PATO
- RUTINA DE TIEMPOS
- DETECCION DE LOS DISPAROS



```
250 SC=SC+(10-VAL(TI$))*5:
    FOR S=10 TO 30:ED=30:
    GOSUB 400:NEXT S:GOTO 50
260 IF SC>HS THEN HS=SC
270 PRINT"[SHIFT+CLR/HOME]
    [CTRL+8]PUNTOS:";SC:
    PRINT"[CTRL+2]MAYOR
    PUNTUACION:";HS
280 PRINT TAB(15);"[2*CRSR
    abajo][CRSR abajo][CTRL+
    4]FIN DEL JUEGO"
290 PRINT TAB(12);"OTRO
    JUEGO?(S/N)"
300 GET A$:IF A$="S" THEN 20
310 IF A$<>"N" THEN 300
320 PRINT"[SHIFT+CLR/HOME]
    [COMM+5]":POKE 53280,14:
    POKE 53281,6:POKE 53272,
    21:END
330 POKE 52,48:POKE 56,48
340 POKE 56334,0:POKE 1,35
350 FOR Z=0 TO 1023:POKE
    12288+Z,PEEK(53248+Z):
    NEXT Z
360 POKE 1,39:POKE 56334,1
370 FOR Z=0 TO 31:READ X:
    POKE 13312+Z,X:NEXT Z:
    POKE 53272,28:RETURN
380 DATA 14,27,63,31,15,7,
    15,31,0,0,0,0,0,192,112,
    188
390 DATA 31,29,30,15,3,1,1,3
    ,206,30,124,248,224,64,
    64,224
400 V=54272:POKE V+24,15:
    POKE V+5,15:POKE V+6,248
410 POKE V+1,S:POKE V+4,17:
    FOR DE=1 TO ED:NEXT DE
420 POKE V+4,0:POKE V+5,0:
    POKE V+6,0:RETURN
```



La línea 10 es una llamada a una subrutina localizada entre las líneas 330 y 370, que establecen los gráficos extra necesarios para el juego.

La línea 330 deja libre algún espacio de memoria para los gráficos. A continuación se copian en RAM los caracteres en ROM, pero antes de hacer esto hay que inhibir las interrupciones a fin de que la copia se realice fielmente (nos ocuparemos de las interrupciones en un artículo posterior). La línea 340 inhibe las interrupciones y deja la ROM de caracteres lista para ser copiada en la línea 350. La ROM de caracteres es copiada y nuevamente se activan las interrupciones en la línea 360.

El gráfico del pato se POKEa en la pantalla por medio de la línea 370 a partir de los DATA de las líneas 380 y 390. Aquí termina la subrutina.

Con el RETURN, el programa vuelve de nuevo a la línea 50, en la que se elige una posición al azar para el pato. También se empieza a contar el tiempo entonces. Seguidamente se incrementa el número de patos, TU, comprobándose que no se han presentado ya más de diez. La cantidad de patos se presenta en la línea 70, y el gráfico del pato se POKEa en pantalla con la línea 80. El tiempo y la puntuación se presentan con las líneas 90 y 100.

La línea 190 comprueba si se aprieta el disparador, saltando a la línea 240 si así es. La siguiente comprobación es la del tiempo. Si el tiempo transcurrido es menor de seis segundos, el programa salta a la línea 80.

La línea 210 define las variables para la subrutina de efectos sonoros de las líneas 400 a 420. S y ED permiten que se utilice la misma subrutina básica para generar varios sonidos diferentes.

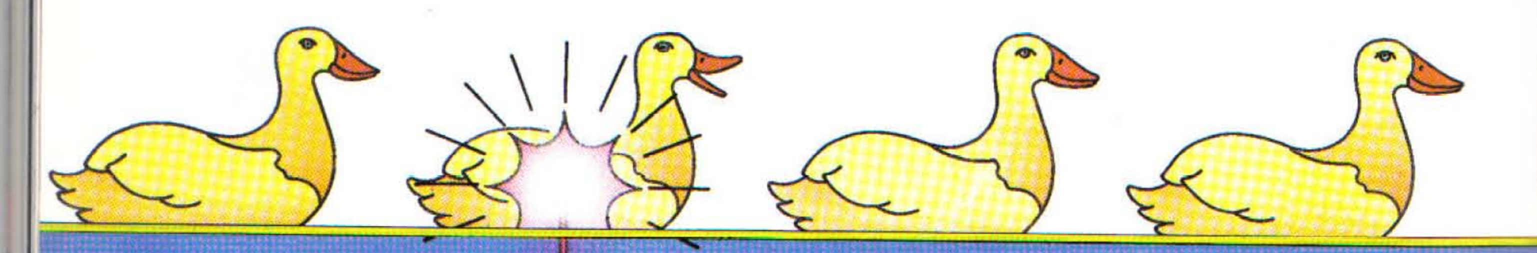
La puntuación actual y la máxima se presentan en la 270, después de que la 250 haya calculado la puntuación actual; el valor exacto de la puntuación depende de lo que el jugador haya tardado en disparar contra el pato. La puntuación máxima se modifica en la línea 260.

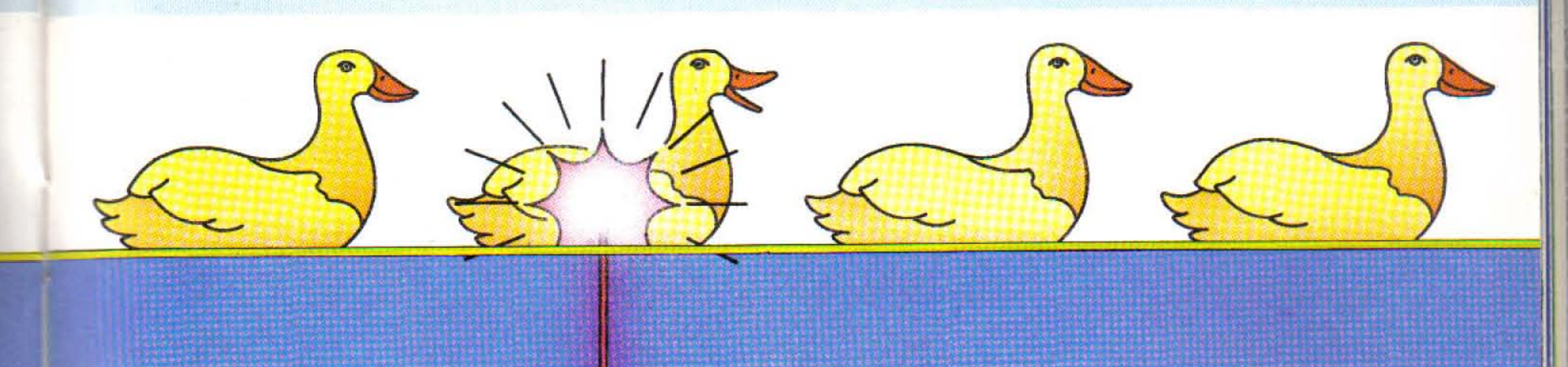
La parte de programa de las líneas 290 a 310 es una rutina del tipo de «¿Quieres jugar otra vez?» como las que ya has visto en otros juegos anteriores. Si el jugador no desea seguir jugando, el color de la pantalla, el color del borde y el modo de gráficos se resetean y el programa finaliza en la línea 320.

El juego se completa para **Vic 20** añadiendo las siguientes líneas a tu rutina de *joystick*:

Teclea para Vic-20

```
10 DATA 14,27,63,31,15,7,15,
   31,0,0,0,0,0,192,112,188
20 DATA 31,29,30,15,3,1,1,3,
   206,30,124,248,224,64,64,
   224
30 FOR Z=0 TO 31:READ X:POKE
   7168+Z,X:NEXT Z
80 D=(7702+INT(RND(1)*19)+1)
   +(INT(RND(1)*20)+1)*22:
   TI$="000000"
90 TU=TU+1:IF TU>10 THEN 360
100 PRINT"[CLR/HOME][2*CRSR
    abajo][CTRL+6][CTRL+9]"
    TAB(7)"PATO:"TU:FOR Z=1
    TO 999:NEXT Z
110 PRINT"[CLR/HOME][2*CRSR
    abajo][CTRL+2][CTRL+9]"
    TAB(7)"[8*ESPACIO]"
120 POKE D,0:POKE D+1,1:POKE
    D+22,2:POKE D+23,3
130 PRINT"[CLR/HOME][CTRL+8]
```





```
    [CTRL+9]"TAB(11)"TM: "
    TI$
140 PRINT"[CLR/HOME][CTRL+9]
    [CTRL+0]PT:[8*ESPACIO]
    [8*CRSR izq]"SC
250 IF F=-1 THEN POKE 36877,
    130:GOTO 300
260 IF TI$<"000005" THEN 120
270 FOR Z=200 TO 127 STEP-1:
    POKE 36875,Z:NEXT Z
280 SC=SC-10
290 POKE D,32:POKE D+1,32:
    POKE D+22,32:POKE D+23,
    32:GOTO 80
300 IF Y<>D THEN SC=SC-10:
    GOTO 330
310 SC=SC+(6-VAL(TI$))*10:
    FOR S=200 TO 127 STEP-1
320 POKE 36865,36+RND(1)*6:
    POKE 36876,S:POKE 36865,
    38:NEXT S
330 POKE 36877,0
340 IF Y=D THEN 80
350 GOTO 120
360 IF SC>HS THEN HS=SC
370 PRINT "[SHIFT+CLR/HOME]
    [CTRL+8][CTRL+9]TUS
    PUNTOS:"SC:PRINT"[CTRL+2]
    [CTRL+9]MAYOR PUNT.:"HS
380 PRINT"[2*CRSR abajo]
    [CTRL+4][6*CRSR dcha]
    [CTRL +9]FIN DE JUEGO"
390 PRINT"[3*CRSR dcha]
    [CTRL+9]OTRA VEZ?
    (S/N)"
400 GET A$:IF A$="S" THEN
    50
410 IF A$<>"N" THEN 400
420 PRINT"[SHIFT+CLR/HOME]
    [CTRL+7]":POKE 36879,27:
    POKE 36869,240
```

El programa del **Vic 20** se parece mucho al del **Commodore 64**.

Las líneas 10 a 30 definen el pato y lo representan en pantalla. Las líneas 80 a 140 son exactamente iguales que las líneas 50 a 100 del programa del **C-64**, y se utilizan para representar el pato y las puntuaciones.

La línea 250 genera un sonido cuando se pulsa el disparador, saltando a continuación a la línea 300. Como el **Vic 20** tiene una retícula de pantalla más pequeña que el **C-64**, la línea 260 establece un límite de tiempo más corto, fijándolo en cinco segundos. La línea 270 genera un sonido de «fuera de tiempo» cuando eres demasiado lento; además en ese caso la línea 280 le resta diez a tu puntuación como penalización por pasarte del tiempo.

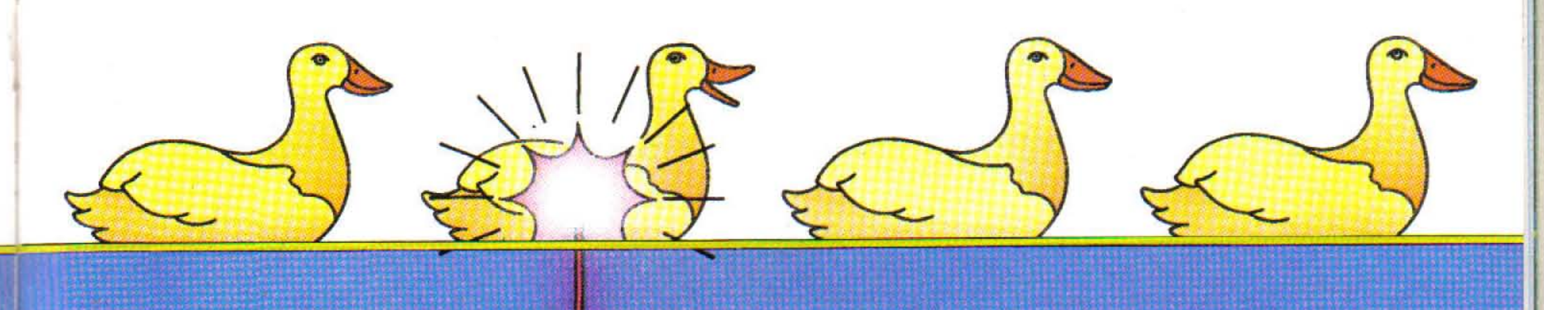
La línea 290 borra el pato de la pantalla y la 300 disminuye la puntuación en diez cuando el jugador falla el dis-

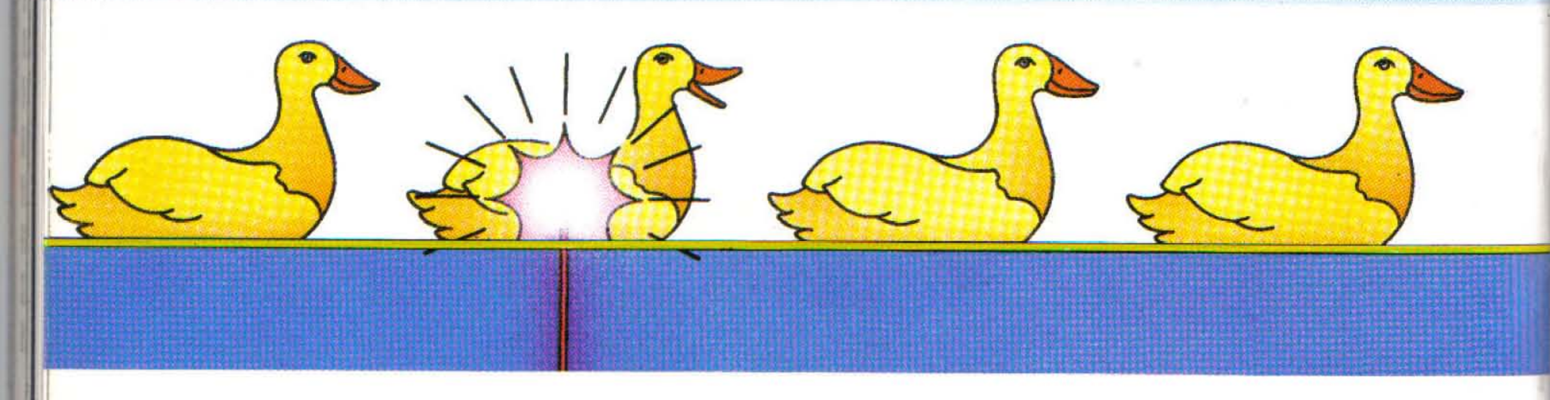
P y R

¿Qué hay que hacer para cambiar el gráfico del pato por un blanco diferente?

La respuesta a esta pregunta varía de unas máquinas a otras. Básicamente se trata de alterar los DATA y de modificar las líneas que impriman el GDU en la pantalla.

Los usuarios de **Commodore** tendréis que modificar los DATA. Si tus nuevos GDU requieren más valores de DATA, también tendrás que cambiar los FOR de la línea 370 (para el **64**) y de la línea 30 (para el **Vic**).





paro. Si el jugador da en el blanco, la línea 310 calcula el tanteo; los puntos resultantes dependen de lo rápidamente que el tirador haya disparado. El bucle FOR ... NEXT de las líneas 310 a 320 genera un sonido de «disparo acertado». La línea 330 suprime el sonido; esta línea también corta el sonido de disparo fallado. Cuando hay un impacto, la línea 340 salta a la línea 80 que vuelve a dibujar de nuevo el pato. En cambio si fallaste el tiro, la línea 350 enviará el programa a la línea 120 que se ocupa de presentar el siguiente pato.

La puntuación actual y la puntuación máxima se presentan en la línea 310, después de que la línea 360 haya actualizado la puntuación máxima en caso de que sea necesario. El resto del programa es una rutina del tipo de «¿quieres jugar otra vez?» a la que sigue la línea 420 que resetea el color de pantalla y el modo.

ENSAMBLADO A MANO (II)

- PRACTICA CON TU ENSAMBLADOR
- RUTINA PARA EL DESPLAZAMIENTO LATERAL DE PANTALLA
- ENTENDIENDO COMO FUNCIONA EL PROGRAMA

Ahora que ya has aprendido a ensamblar a mano, el mundo del ordenador doméstico se convierte en un vivero para tí. Posees un buen conocimiento del BASIC, puedes manejar además el código máquina y la traducción de lenguaje ensamblador a código màquina no te presentará problemas.

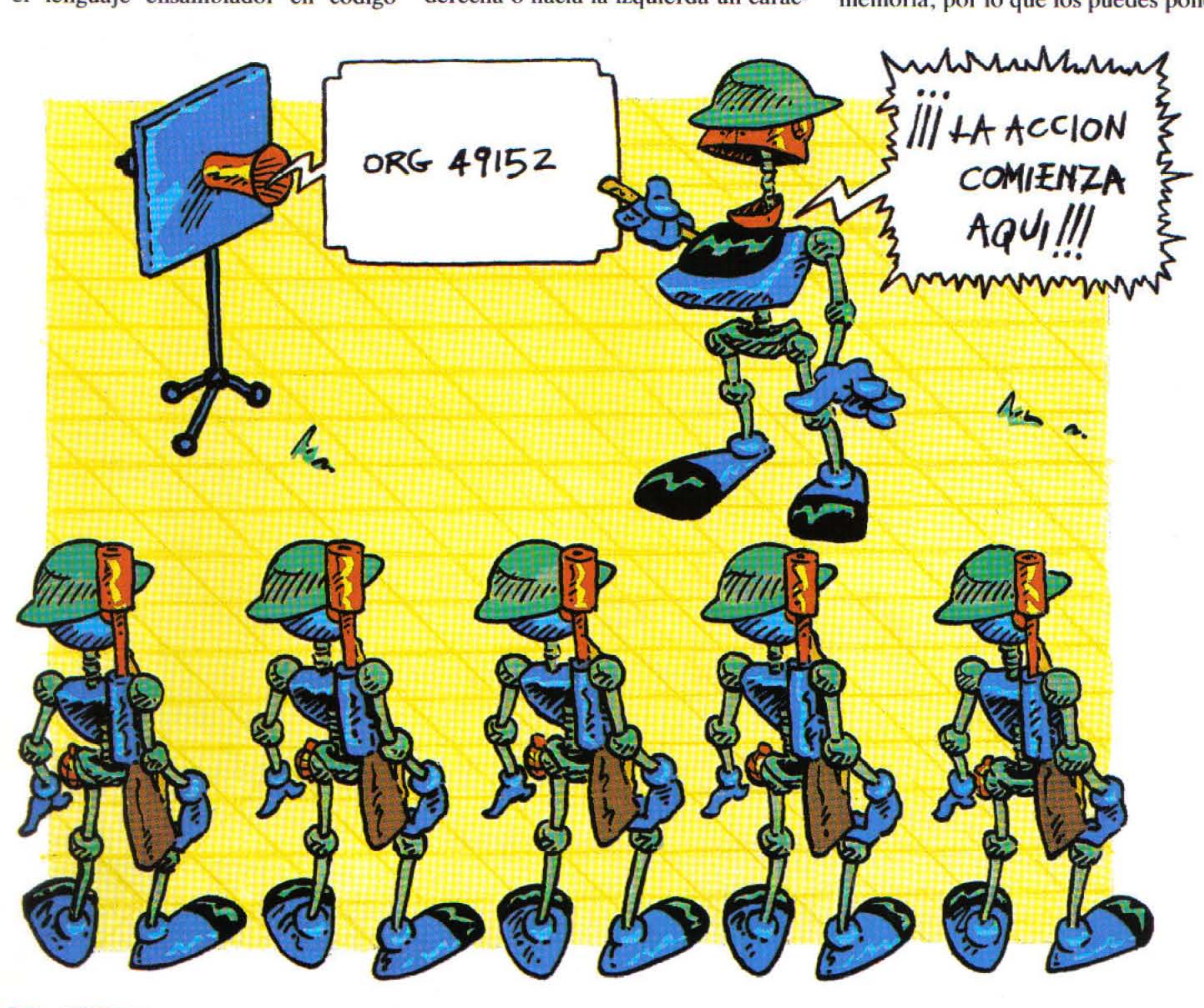
En la primera parte de este artículo ya has podido ver cómo se hace el ensamblado a mano y cómo se convierte el lenguaje ensamblador en código máquina, por lo que ya puedes introducirlo en tu ordenador y ejecutarlo.

Cuando ya domines este tema, estarás en condiciones de usar estas técnicas en algunas rutinas útiles.

SCROLLING LATERAL

Los siguientes programas en lenguaje ensamblador sirven para desplazar lateralmente la pantalla hacia la derecha o hacia la izquierda un carácter cuadrado. Ensámblalos a mano e introduce el código máquina en tu ordenador, con ayuda de tu programa mónitor de código máquina. Si quieres almacenar estos programas con vistas a su posible utilización futura, no te olvides de ponerlos en secciones distintas de memoria, para el caso de que quieras usar ambos al mismo tiempo.

Los dos programas funcionan independientemente de su situación en la memoria, por lo que los puedes poner



en cualquier posición que resulte conveniente para tu máquina.

Si quieres hacer un *scrolling* de pantalla hacia la derecha o hacia la izquierda, de más de un caracter de extensión, puedes llamar a esta rutina desde un bucle FOR ... NEXT. O si quieres hacerlo cada vez que se pulse una determinada tecla, puedes usar un programa en BASIC que utilice GET$.

Esta rutina en lenguaje ensamblador hace un *scrolling* de pantalla hacia la izquierda. Aquí la tienes traducida en lenguaje máquina.

Teclea para Commodore-64

```
        LDA #$00        A9 00
        STA $FB         85 FB
        LDA #$04        A9 04
        STA $FC         85 FC
        LDA #$00        A9 00
        STA $FE         85 FE
OTRVEZ  LDY #$00        A0 00
        LDA ($FB),Y     B1 FB
        STA $FD         85 FD
        LDY #$01        A0 01
BUCLE   LDA ($FB),Y     B1 FB
        DEY                88
        STA ($FB),Y     91 FB
        INY                C8
        INY                C8
        CPY #$28        C0 28
        BNE BUCLE       D0 28
        LDY #$27        A0 27
        LDA $FD         A5 FD
        STA ($FB),Y     91 FB
        LDA $FB         A5 FB
        ADC #$27        69 27
        STA $FB         85 FB
        BCC SALTO       90 02
        INC $FC         E6 FC
SALTO   INC $FE         E6 FE
        LDX $FE         A6 FE
        CPX #$19        E0 19
        BNE OTRVEZ      D0 D5
        RTS                60
```

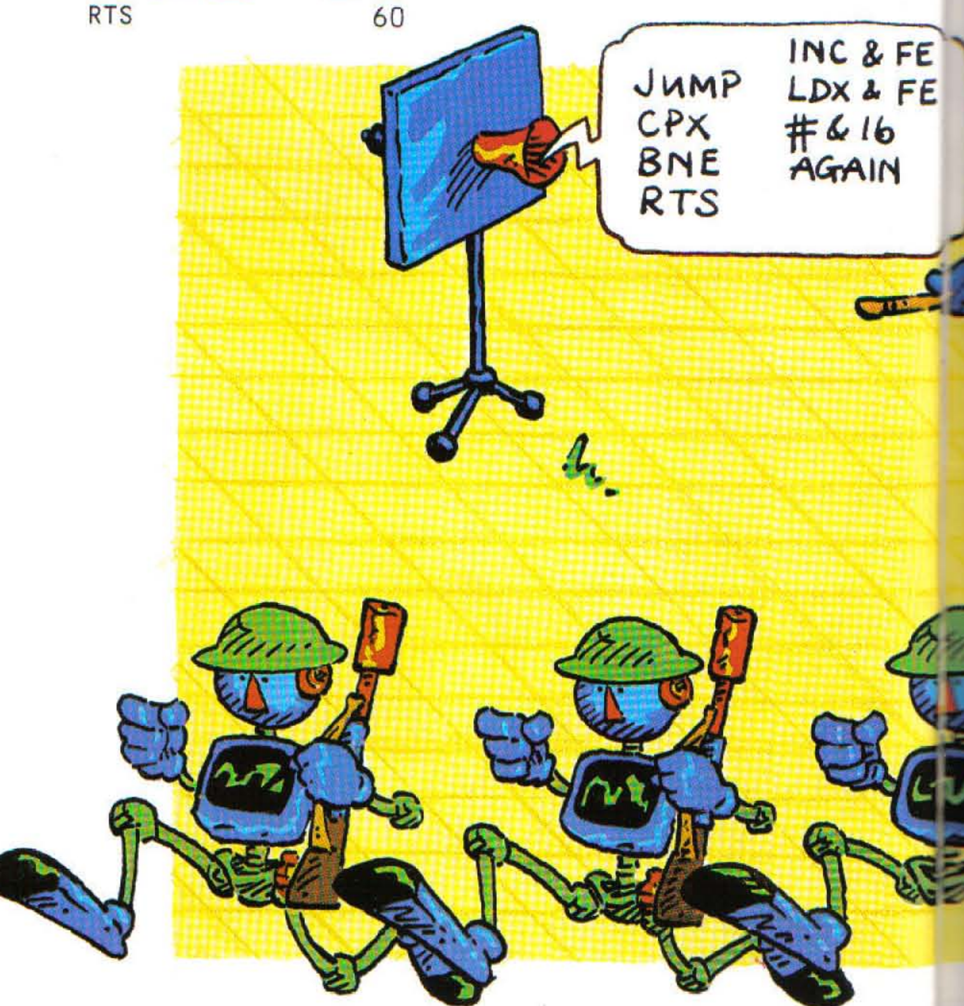
Teclea para Vic-20

```
        LDA #$00        A9 00
        STA $FB         85 FB
        LDA #$1E        A9 1E
        STA $FC         85 FC
        LDA #$00        A9 00
        STA $FE         85 FE
OTRVEZ  LDY #$00        A0 00
        LDA ($FB),Y     B1 FB
        STA $FD         85 FD
        LDY #$01        A0 01
BUCLE   LDA ($FB),Y     B1 FB
        DEY                88
        STA ($FB),Y     91 FB
        INY                C8
        INY                C8
        CPY #$16        C0 16
        BNE BUCLE       D0 28
        LDY #$15        A0 15
        LDA $FD         A5 FD
        STA ($FB),Y     91 FB
        LDA $FB         A5 FB
        ADC #$15        69 15
        STA $FB         85 FB
        BCC SALTO       90 02
        INC $FC         E6 FC
SALTO   INC $FE         E6 FE
        LDX $FE         A6 FE
        CPX #$17        E0 17
        BNE OTRVEZ      D0 D5
        RTS                60
```

FUNCIONAMIENTO

LDA #$00 carga el valor 0 en el acumulador, y **STA $FB** lo almacena en la dirección de memoria $00FB. Análogamente las instrucciones **LDA #$04** y **STA $FC** almacenan el valor $04 en la dirección de memoria $00FC utilizando como intermediario el acumulador; todo esto en el **C-64**, en el **Vic** las instrucciones **LDA #$1E** y **STA $FC** sirven para almacenar, a través del acumulador, el valor $1E en la dirección de memoria $00FC. No hay un comando que permita almacenar directamente un valor en una determinada posición de memoria.

$0400 es la dirección de la primera posición de pantalla del **C-64**, mientras que $1E es la dirección de la primera posición de pantalla en el **Vic**. $00FB y $00FC son dos posiciones de la parte de usuario en la página cero.



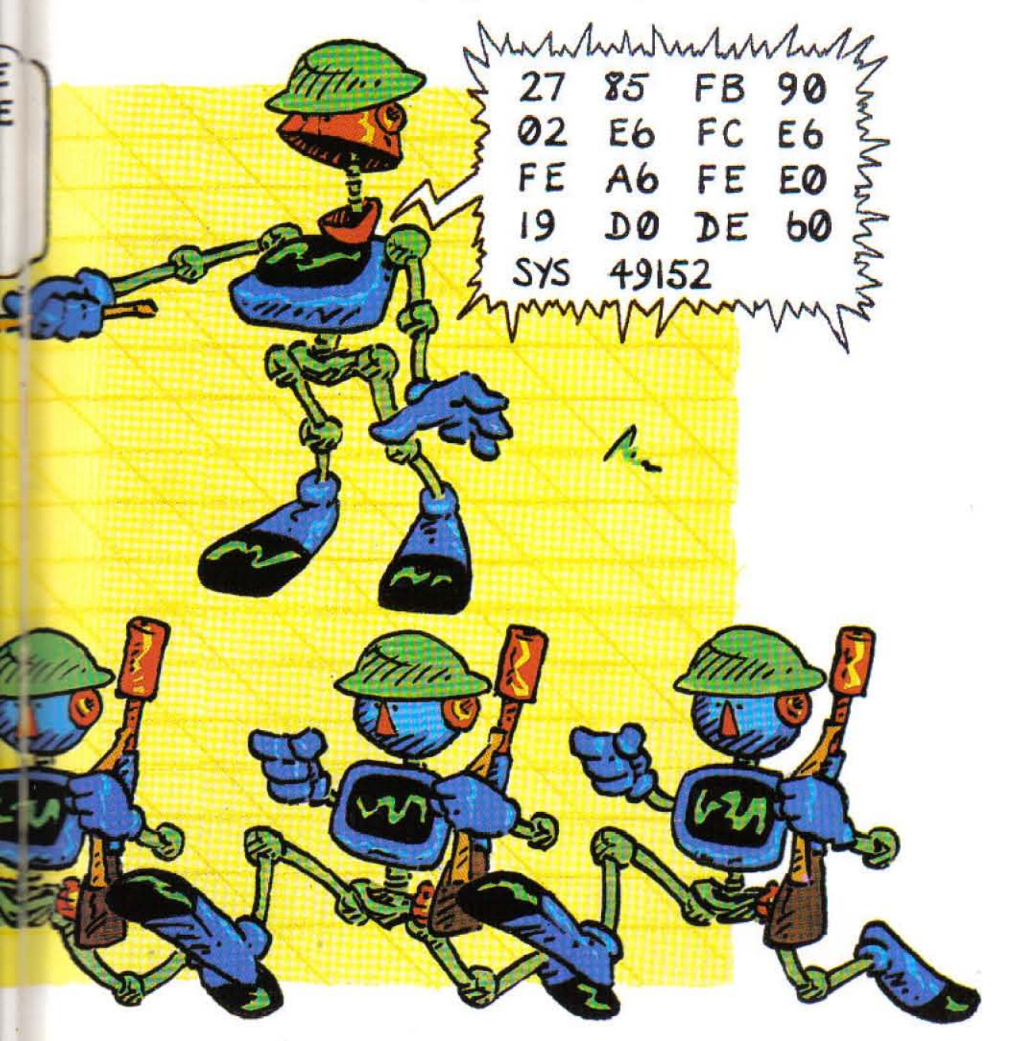
La página cero contiene las direcciones que pueden especificarse mediante un solo byte.

Las instrucciones **LDA #$00** y **STA $FE** cargan el valor 0 en la dirección de memoria $00FE. Utilizaremos esta dirección como contador.

LDY #$00 carga el registro de índice Y con el valor del primer desplazamiento, que evidentemente es cero, ya que el programa de *scrolling* tiene que empezar al principio de la pantalla. La instrucción **LDA ($FB),Y** carga el acumulador con el contenido de la dirección de memoria dada por $00FB y el siguiente byte $00FC, más un desplazamiento que viene dado por el contenido del registro Y. Los bytes $00FB y $00FC están apuntando a $0400 en el **C-64** y a $1E00 en el **Vic**, que es el principio de la pantalla, siendo el desplazamiento nulo, por lo que esta instrucción carga el contenido del primer carácter en el acumulador y seguidamente **STA $FD** lo almacena en $00FD.

A continuación **LDY #$01** carga el registro Y con 1. **LDA ($FB),Y** carga el acumulador con el contenido de la dirección de memoria dada por $00FB y $00FC, más un desplazamiento que nuevamente viene dado por el contenido del registro Y. Pero esta vez el registro Y no contiene un 0 sino un 1. Por ello esta instrucción carga el acumulador con el contenido de $0401 en el **C-64** y con el de $1E01 en el **Vic**, que corresponden a la segunda posición de pantalla.

Con **DEY** se decrementa el registro Y mientras que **STA ($FB),Y** almacena el contenido del acumulador en la dirección dada por $00FB, $00FC y el desplazamiento que haya en Y. Cuando en el proceso Y ha sido decrementado en una unidad, tiene el efecto de desplazar un lugar el contenido da cada posición de memoria.



El registro Y se incrementa entonces dos veces por medio de la instrucción **INY**, quedando preparado para apuntar a la siguiente posición de pantalla. **CPY #$28** compara el contenido del registro Y con $28 en hexadecimal, o lo que es lo mismo 40 en decimal, en el **C-64**. La pantalla del **C-64** tiene 40 columnas. Con **CPY #$28** se activa el indicador de cero si el contenido del registro Y es 40 en decimal.

CPY #$16 compara el contenido del registro Y con $16 en hexadecimal o 22 en decimal, en el **Vic**. La pantalla del **Vic** tiene 22 columnas. Con **CPY #$16** se activa el indicador de cero si el contenido del registro Y es 22 en decimal.

BNE BUCLE examina el indicador de cero. Si no ha sido activado, el micro salta hacia atrás hasta donde encuentre la etiqueta **BUCLE** y vuelve a empezar de nuevo esta sección del programa.

BNE significa saltar si el resultado no es igual, por lo que continuará enviando al micro hacia atrás al bucle mientras el registro Y va contando de 1 a 40 en el **C-64** y de 1 a 22 en el **Vic**, desplazando cada uno de los caracteres de un espacio. Cuando el contenido del registro Y llega al valor 40 en el **C-64** o a 22 en el **Vic**, la condición **BNE** deja de cumplirse y el micro pasa a ejecutar la siguiente instrucción.

LDY#$27 carga el registro Y con 27 en hexadecimal o 39 en decimal en el **C-64**, mientras que en el **Vic LDY #$15** carga el registro Y con el valor 21. **LDA $FD** carga el acumulador con el contenido de $00FD. Si miras hacia atrás, verás que $00FD contiene el primer carácter.

STA ($FB),Y carga el contenido del acumulador en las direcciones de memoria dadas por $00FB y $00FC ($0400 en el **C-64** y $1E00 en el **Vic**) y por el desplazamiento del registro Y (27 hexadecimal en el **64** y 15 hexadecimal en el **Vic**). Así esta instrucción sitúa el contenido del primer carácter en la dirección de memoria $0427 para el **C-64** y $1E15 para el **Vic**, que corresponde al último carácter de la primera fila pasando así el primer carácter del primero al último lugar.

LDA $FB carga el acumulador con

el contenido de la posición de memoria $00FB, a continuación **ADC #$27** le suma 39 en el **C-64**, mientras que en el **Vic ADC #$15** suma 21. El resultado se coloca en la dirección de memoria $00FB por medio de la instrucción **STA $FB**. Con ello queda el programa en condiciones de manejar la siguiente fila de la pantalla.

BCC significa salto si el acarreo es cero, mientras que **ADC** significa suma con acarreo. Así, si en la instrucción **ADC** no se produce desbordamiento del acumulador de ocho bits, y en consecuencia no se activa el indicador de acarreo, la instrucción **BCC** hace que el micro salte a la instrucción **INC $FE**.

Pero si la instrucción **ADC** hace que el acumulador se desborde y el indicador de acarreo se activa, la instrucción a la que salta el micro es la **INC $FC**. Simplemente lo que hace es incrementar el contenido de la dirección de memoria $00FC que contiene el byte alto del puntero de posición de pantalla. Así se asegura que el puntero se incrementa adecuadamente y no se ha perdido nada en el curso de las operaciones anteriores.

La instrucción **INC $FE** incrementa el contador de $00FE. A continuación **LDX $FE** carga el contenido de $00FE en el registro X de forma que **CPX #$19** pueda compararlo con el valor 25 que es el número de filas de la pantalla del **C-64**; en el **Vic**, la comparación se hace con **CPX #$17**, siendo 23 el valor decimal con el que se compara, ya que este es precisamente el número de filas de la pantalla del **Vic**.

Si el valor del contador de $00FE es menor que 25 en el **C-64** o que 23 en el **Vic**, la instrucción **CPX** no activa el indicador de cero, por lo que entra en acción la instrucción **BNE** y envía de nuevo al micro al principio de la rutina para actuar sobre la siguiente fila. Pero en cambio si el contador de $00FE ha llegado a 25 en el **C-64** o a 23 en el **Vic**, no se satisface la condición para que el microprocesador pase a la próxima instrucción.

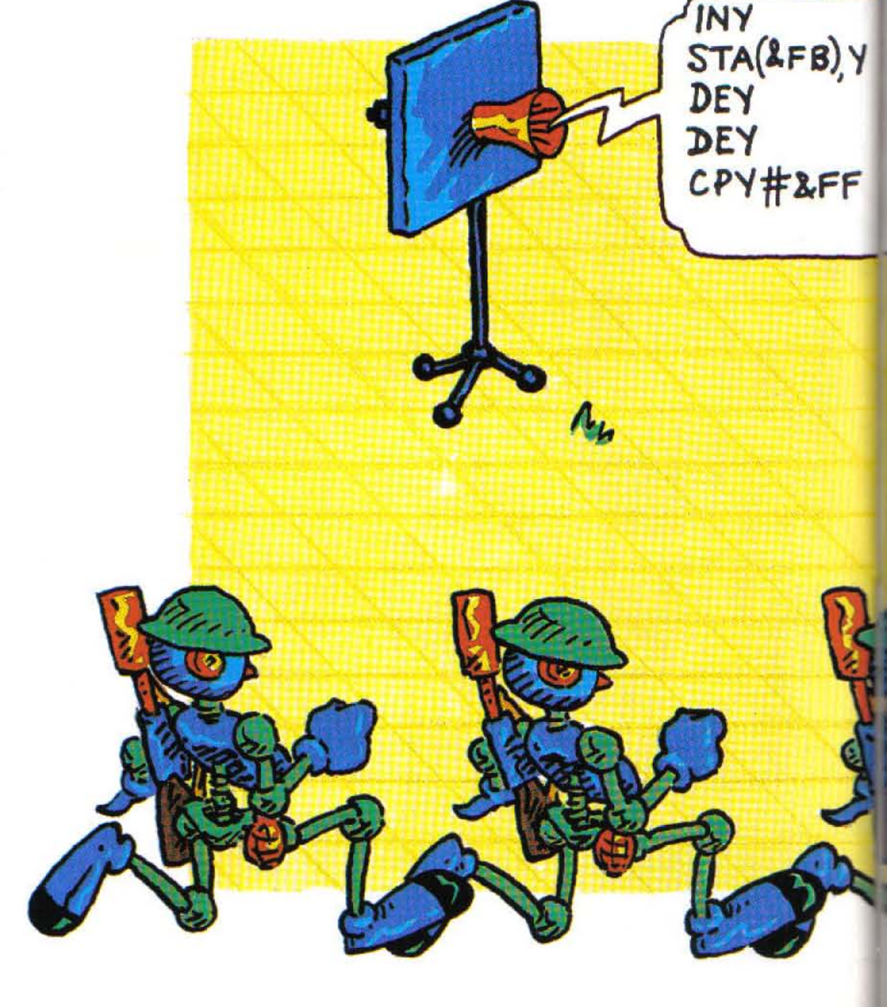
La instrucción **RTS** le dice al microprocesador que vuelva al BASIC. Toda subrutina en código máquina tiene que terminar con **RTS** ya que de no ser así el microprocesador empezará a recorrer la memoria intentando ejecutar todo lo que pudiera encontrarse, y terminará por quedarse atascado. Si tienes suerte, puede ser que consigas salvar tu programa por medio de RUN/STOP y RESTORE .

Como puedes observar el programa sólo hace *scrolling* de caracteres y no de colores. De esta manera podrías llenar la pantalla con un color de fondo diferente del que tenías originalmente, utilizando únicamente el *scrolling*.

El programa en lenguaje ensamblador que sigue a continuación hace un *scrolling* de pantalla hacia la derecha. Prueba a ensamblarlo a mano:

Teclea para Commodore-64

```
        LDA #$00       A9 00
        STA $FB        85 FB
        LDA #$04       A9 04
        STA $FC        85 FC
        LDA #$00       A9 00
        STA $FE        85 FE
OTRVEZ  LDY #$27       A0 27
        LDA ($FB),Y    B1 FB
        STA $FD        85 FD
        LDY #$26       A0 26
BUCLE   LDA ($FB),Y    B1 FB
        INY               C8
        STA ($FB),Y    91 FB
        DEY               88
        DEY               88
        CPY #$FF       C0 FF
        BNE BUCLE      D0 28
        LDY #$00       A0 00
        LDA $FD        A5 FD
        STA ($FB),Y    91 FB
        LDA $FB        A5 FB
        ADC #$27       69 27
        STA $FB        85 FB
        BCC SALTO      90 02
        INC $FC        E6 FC
SALTO   INC $FE        E6 FE
        LDX $FE        A6 FE
        CPX #$19       E0 19
        BNE OTRVEZ     D0 D5
        RTS               60
```



Teclea para Vic-20

```
         LDA #$00        A9 00
         STA $FB         85 FB
         LDA #$1E        A9 1E
         STA $FC         85 FC
         LDA #$00        A9 00
         STA $FE         85 FE
OTRVEZ   LDY #$15        A0 15
         LDA ($FB),Y     B1 FB
         STA $FD         85 FD
         LDY #$14        A0 14
BUCLE    LDA ($FB),Y     B1 FB
         INY                C8
         STA ($FB),Y     91 FB
         DEY                88
         DEY                88
         CPY #$FF        C0 FF
         BNE BUCLE       D0 28
         LDY #$00        A0 00
         LDA $FD         A5 FD
         STA ($FB),Y     91 FB
         LDA $FB         A5 FB
         ADC #$15        69 15
         STA $FB         85 FB
         BCC SALTO       90 02
         INC $FC         E6 FC
SALTO    INC $FE         E6 FE
         LDX $FE         A6 FE
         CPX #$16        E0 16
         BNE OTRVEZ      D0 D5
         RTS                60
```

Como habrás observado este programa funciona de la misma forma que el programa de *scrolling* hacia la izquierda, excepto que en la rutina de desplazamiento de línea ahora incrementas donde antes decrementabas y decrementas donde antes incrementabas.

Los saltos son iguales que antes, por lo que podrás comprobar que están bien dimensionados sin más que examinar los listados de los anteriores programas en código máquina. No te olvides de que tienes que calcular el salto desde el final de la instrucción de salto lo que significa que tienes que incluir en el mismo el byte de la instrucción del propio salto.

Teclea para Commodore-64

```
1 REM**EJEMPLO CON CARGA
  DESDE BASIC DE RUTINAS
  EN C.M.
5 REM**CARGADOR BASIC PARA
  RUTINA DE SCROLLING IZQ.
10 FOR S=0 TO 55
20 READ A
30 POKE 49152+S,A
40 NEXT S
100 DATA 169,0,133,251,169,
    4,133,252,169,0
110 DATA 133,254,160,0,177,
    251,133,253,160,1
120 DATA 177,251,136,145,
    251,200,200,192,40,208
130 DATA 245,160,39,165,253,
    145,251,165,251,105,39
140 DATA133,251,144,2,230,
    252,230,254,166,254
150 DATA 224,25,208,213,96
200 REM**RUTINA SCROLLING
    DERECHA
210 FOR S=0 TO 55
220 READ A
230 POKE 49216+S,A
240 NEXT S
300 DATA 169,0,133,251,169,
    4,133,252,169,0,133,254
310 DATA 160,39,177,251,133,
    253,160,38,177,251,200
320 DATA 145,251,136,136,192
    ,255,208,245,160,0,165
330 DATA 253,145,251,165,251
    ,105,39,133,251,144,2
340 DATA 230,252,230,254,166
    ,254,224,25,208,213,96
500 REM**SYS 49152 DESPLAZA
    1 ESPACIO A LA IZQ.
510 REM**SYS 49216 DESPLAZA
    1 ESPACIO A LA DER.
1000 FOR G=1 TO 1000:NEXT G
1010 D=40*RND(1):E=40*RND(1)
     :R=300*RND(1)
1020 FOR X=1 TO D
1030 SYS49152
1040 NEXT X
1050 FOR Y=1 TO E
1060 SYS49216
1070 NEXT Y
1080 FOR T=1 TO R:NEXT T
1090 GOTO 1000
```

¿MONITOR DE VIDEO O TELEVISOR?

- COMO FUNCIONAN
- TIPOS DE ENTRADAS DE SEÑAL
- MONOCROMO Y COLOR
- DEFINICION DE LA IMAGEN

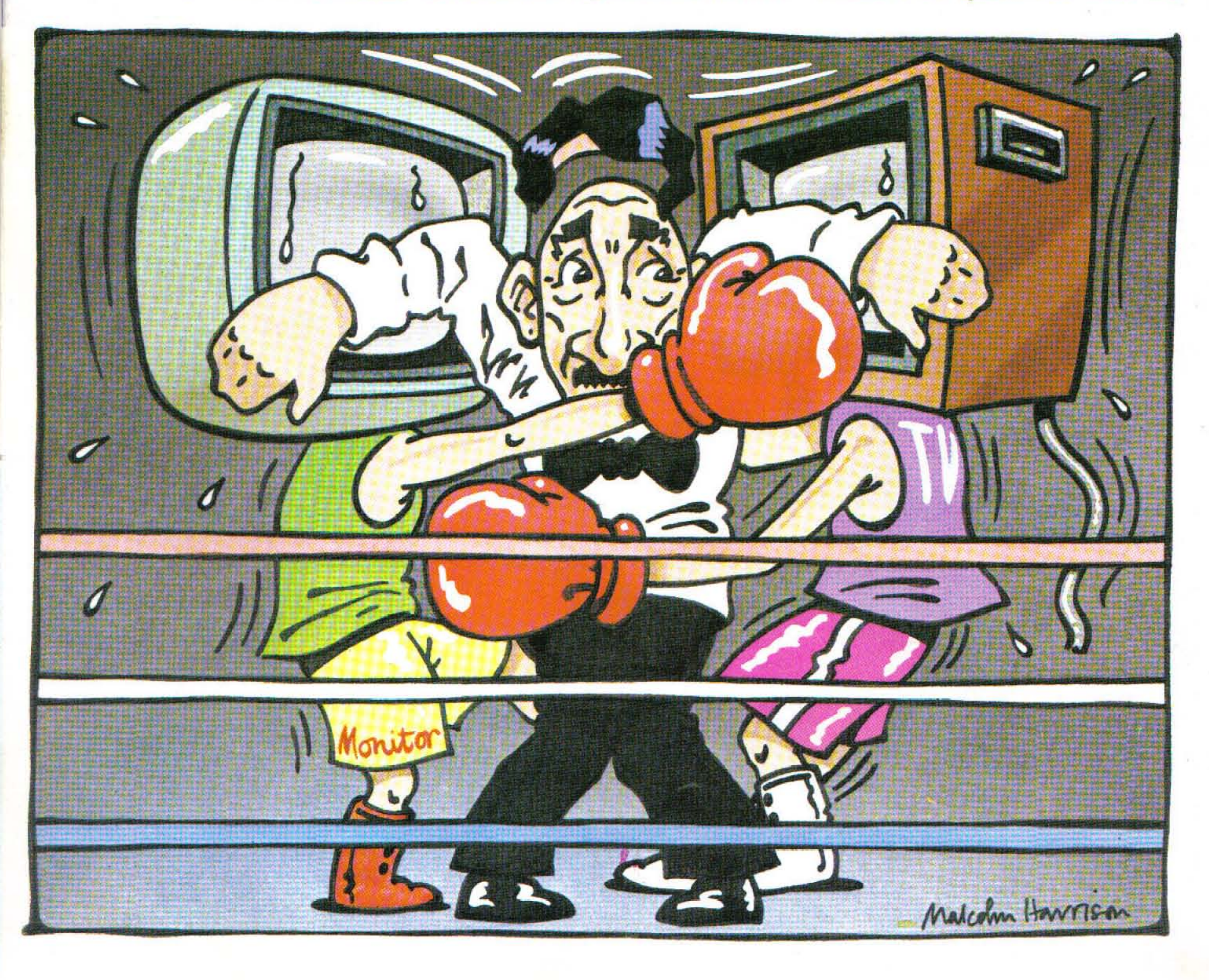
Puede ser difícil realizar una elección entre un televisor de calidad y un monitor de video, pero en muchos casos no cabe más que una opción. En este artículo dedicamos nuestra atención a algunas de las diferencias generales entre los dos tipos.

La pantalla es un elemento vital de cualquier ordenador, aunque muchas veces se la considera como una especie de mula de carga en vez del sofisticado elemento electrónico que en realidad es. La pantalla es el principal canal de comunicaciones entre el ordenador y nosotros. En el otro sentido la información fluye desde el usuario hacia el ordenador, normalmente a través del teclado, representando también la pantalla en este caso una réplica o eco de la información que se tecleó.

Si estás en una situación tan afortunada como para tener que escoger una nueva pantalla, o simplemente deseas aprovechar lo mejor posible la que tienes, te resultará conveniente, si no esencial, tener algunas nociones sobre la forma en que funciona una pantalla y la forma en que el ordenador controla la salida de video. En particular esto último es algo que afectará a tu elección.

PRINCIPIOS BASICOS

Aunque hay varias formas diferentes de transferir la información desde el ordenador a la pantalla, una cosa es común a todos los tipos de pantallas utilizados en aplicaciones tanto co-



merciales como industriales o domésticas: el tubo de rayos catódicos, también conocido por sus iniciales inglesas CRT. La base del funcionamiento de los tubos de rayos catódicos reside en el hecho de que ciertos compuestos de fósforo emiten luz cuando son excitados por un haz de electrones.

Si se dispara un chorro de electrones con un cañón electrónico sobre una lámina de vidrio cubierta con fósforo, cada vez que se produzca un disparo el fósforo brillará. Si se desplaza el cañón de forma que el chorro de electrones vaya recorriendo toda la pantalla, será toda la superficie de la misma lo que emitirá brillo al estar sometida al bombardeo de electrones. Si se hace que el haz de electrones se mueva, por ejemplo, describiendo la figura de un ocho, aparecerá en la pantalla la figura de un ocho.

Un tubo de rayos catódicos es un cono de vidrio cerrado en cuyo interior se ha hecho el vacío, provisto de un cañón electrónico en el extremo próximo al vértice y con su base cubierta de fósforo.

Imaginemos un trozo de cartulina en el que se ha recortado la forma de un hombre, interponiéndose entre el cañón de electrones y la pantalla. La pantalla sólo brillará en los puntos en que el chorro de electrones no haya sido bloqueado por la cartulina. En consecuencia, la zona de pantalla cuyo fósforo brilla tendrá una forma parecida a la de un hombre. Se podría conseguir el mismo efecto interrumpiendo el haz electrónico (apagando el cañón) en los instantes adecuados, es decir cuando pasa por los puntos deseados de la pantalla.

Esta es exactamente la forma en que funcionan los televisores y la mayoría de los monitores, a excepción de algunos de construcción especial. Un cañón electrónico barre todo el ancho de la pantalla 625 veces para crear 625 líneas. (En realidad barre primero 313 líneas y a continuación rellena los 312 huecos que había dejado entre dichas líneas.)

El sistema no tiene partes mecánicas móviles, el desplazamiento del haz de electrones se consigue por medio de campos magnéticos. Por eso está recomendado no dejar materiales magnéticos —*diskettes* o cintas— cerca de un televisor. Para formar las imágenes, se conmuta el cañón de electrones encendiéndolo y apagándolo. La información que contiene la señal recibida por el televisor o el monitor, precisamente sirve para decirle al cañón de electrones cuándo tiene que emitirlos o no.

En lugar de recorrer toda la pantalla en lo que se llama un barrido por rastreo, algunos monitores producen la imagen guiando el haz de electrones por la pantalla como si fuera un lápiz, siguiendo las líneas de la imagen. Aunque se consiguen así resultados de excelente calidad, se trata de un procedimiento mucho más lento para usos generales. Aún suponiendo que el haz de electrones se desplazara por la pantalla a la nada despreciable velocidad de 40000 kilómetross por hora, una imagen de cierta complejidad aparecería ante el ojo humano como altamente insatisfactoria.

PERSISTENCIA DE LA VISION

Así se llama una de las características del ojo humano que en otras circunstancias podría parecer un defecto, pero que en este caso permite la utilización del tubo de rayos catódicos como instrumento de comunicación visual.

El ojo humano tiene la propiedad de mantener cualquier imagen de las que se forman sobre su retina durante un tiempo de aproximadamente 1/25 de segundo. El cañón de electrones que hay en la parte trasera de los tubos de rayos catódicos de los televisores domésticos y de la mayoría de los monitores, barre la superficie de la pantalla 625 veces en 1/50 de segundo. Esto significa que se forma una nueva semi-imagen o «trama» sobre la pantalla cincuenta veces por segundo. De esta forma aparece sobre la pantalla una imagen nueva antes de que se haya desvanecido la antigua de nuestro ojo. Por eso podemos ver una imagen que se mueve con suavidad.

Si nuestro ojo fuera capaz de retener la imagen solamente durante una centésima de segundo, la imagen de la pantalla nos parecería que parpadeaba de forma insoportable.

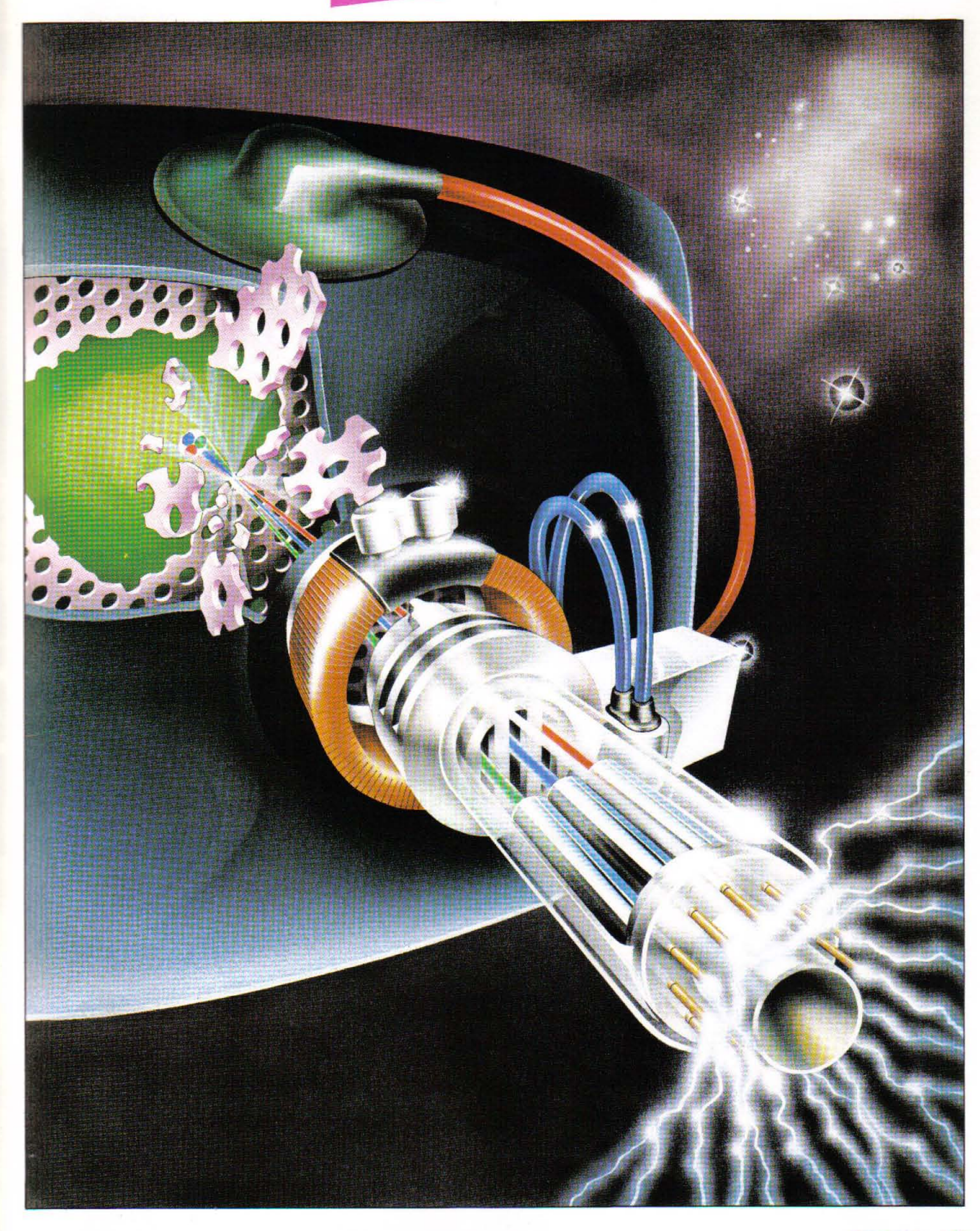
COLOR

El principio básico empleado es el mismo para televisores en blanco negro y color. La única diferencia entre ambos es que sólo se requiere un cañón de electrones y un tipo de recubrimiento de fósforo en el caso del monocromo, mientras que en el caso del color hacen falta tres tipos de fósforos diferentes en el recubrimiento de la pantalla y normalmente tres cañones electrónicos (hay televisores que tienen un solo cañón, pero que emiten tres haces de electrones).

El color se debe a que uno de los fósforos del recubrimiento produce luz roja, otro luz verde y el otro luz azul. Uno de los chorros de electrones activa el fósforo rojo, otro el fósforo verde y el tercero el fósforo azul. Combinando el rojo, verde y azul con diferentes intensidades se pueden obtener todos los demás colores e intensidades.

ENTRADA DE LA SEÑAL

Las pantallas, ya sean de un televisor o de un monitor, de color o de blanco y negro, pueden aceptar tres clases de señales de entrada. En primer lugar está la entrada normal de televisión enviada por las emisoras públicas, que es la señal que acepta el televisor de tu casa y produce en la pantalla las imágenes de televisión que estás acostumbrado a ver. Por otra parte está la llamada señal compuesta de videofrecuencia o vídeo compuesto. Esta es una de las señales que se emplea con más frecuencia en los monitores de vídeo y en la actualidad ya hay algunos televisores que tienen prevista una toma para aceptar dichas señales. La señal de vídeo compuesto contiene la información de la imagen y los impulsos de sincronismo. Esta última hace que el haz de electrones se sitúe en el lugar adecuado en cada momento.



AMSTRAD CPC - 464

AMSTRAD



ORDENADORE

SERIE CPC

UNIDAD CENTRAL. MEMORIAS.
• Microprocesador Z80A - 64K RAM ampliables - 32K ROM ampliables
• TECLADO • Teclado profesional con 74 teclas en 3 bloques - Hasta 32 teclas programables - Teclado redefinible
• PANTALLA • Monitor RGB verde (12") o color (14")

	Normal	Alta Res.	Multicolor
Col × líneas	40 × 25	80 × 25	20 × 25
Colores	4 de 27	2 de 27	16 de 27
Puntos	320 × 200	640 × 200	160 × 2

— Se pueden definir hasta 8 ventanas de texto y 1 de gráficos **• SONIDO**
• 3 canales de 8 octavas moduladas independientemente - Altavoz interno regulable - Salida estéreo **• BASIC**
• Locomotive BASIC ampliado en ROM - Incluye los comandos AFTER y EVERY para control de interrupciones

AMSTRAD CPC 464

CASSETTE • Cassette incorporada con velocidad de grabación (1 ó 2 Kbaudios) controlada desde Basic **• CONECTORES**
• Bus PCB multiuso, Unidad de Disco exterior, paralelo Centronics, salida estéreo, joystick, lápiz óptico, etc.
• SUMINISTRO • Ordenador con monitor verde o color - 8 cassettes con programas - Libro "Guía de Referencia BASIC para el programador" - Manual en castellano - Garantía Oficial AMSTRAD ESPAÑA.

TODO POR **59.900 Pts.** (monitor verde)
90.900 Pts. (monitor color)

AMSTRAD CPC 6128

UNIDAD DE DISCO • Unidad incorporada para disco de 3" con 180K por cara **• SISTEMAS OPERATIVOS**
• AMSDOS, CP/M 2.2, CP/M Plus (3.0)
• CONECTORES • Bus PCB multiuso, paralelo Centronics, cassette exterior, 2.ª Unidad de Disco, salida estéreo, joysticks, lápiz óptico, etc.
• SUMINISTRO • Ordenador con monitor verde o color - Disco con CP/M 2.2 y lenguaje DR. LOGO - Disco con CP/M Plus y utilidades - Disco con 6 programas de obsequio - Manual en castellano - Garantía Oficial AMSTRAD ESPAÑA.

TODO POR **84.900 Pts.** (monitor verde)
119.900 Pts. (monitor color)

D PCW - 8256

AMSTRAD CPC - 6128

ES AMSTRAD ¡¡Increíble!!

AMSTRAD PCW 8256

UNIDAD CENTRAL. MEMORIAS
• Microprocesador Z80A - 256K RAM de las que 112K se utilizan como disco RAM
• **TECLADO** • Teclado profesional en castellano (ñ, acento...) de 82 teclas
• **PANTALLA** • Monitor verde de alta resolución - 90 columnas × 32 líneas de texto • **UNIDAD DE DISCO** • Disco de 3" y 173K por cara - Opcionalmente, 2.ª Unidad de Disco de 1 Mbyte integrable
• **SISTEMA OPERATIVO** • CP/M Plus de Digital Research • **IMPRESORA** • Alta calidad (NLQ) a 20 c.p.s. - Calidad estándar a 90 c.p.s. - Papel continuo u hojas sueltas - Alineación automática del papel - Caracteres normales, comprimidos, expandidos, control del paso de letra (normal, cursiva, negrita, subíndices, superíndices, subrayado, etc).
• **OPCIONES** • Kit de Ampliación a 512K RAM y 2.ª Unidad de Disco - Interface Serie RS 232C y paralelo Centronics • **SUMINISTRO** • Ordenador completo con teclado, pantalla, Unidad de Disco e Impresora - Discos con el procesador de Texto LocoScript, CP/M Plus, Mallard, BASIC, DR. LOGO y diversas utilidades - Manuales en castellano - Garantía Oficial AMSTRAD ESPAÑA.

TODO POR 129.900 Pts.



Los más prestigiosos paquetes de **Software Profesional,** en formato AMSTRAD... a **"precios AMSTRAD"**

Existe también la versión **PCW 8512** con **512K RAM** y la 2.ª Unidad de Disco de 1 Mbyte incorporada. **PVP. 174.900 Pts.**
* El **PCW 8256** puede utilizarse como terminal y en comunicaciones.

El I.V.A. no está incluido en los precios.

NOTA: Es muy importante verificar la garantía del aparato ya que sólo **AMSTRAD ESPAÑA** puede garantizarle la ordenada reparación y sobre todo materiales de repuesto oficiales (Monitor, ordenador, cassette o unidades de discos).

AMSTRAD ESPAÑA

Avda. del Mediterráneo, 9. Tels. 433 45 48 - 433 48 76. 28007 MADRID

Delegación Cataluña: Tarragona, 110 - Tel. 325 10 58. 08015 BARCELONA

Por último, está la llamada entrada RGB. Las letras corresponden a rojo, verde y azul (Red, Green, Blue). Es la forma más directa y más precisa de especificar la información de color. La información relativa a cada color se introduce directamente y por separado desde el ordenador al monitor.

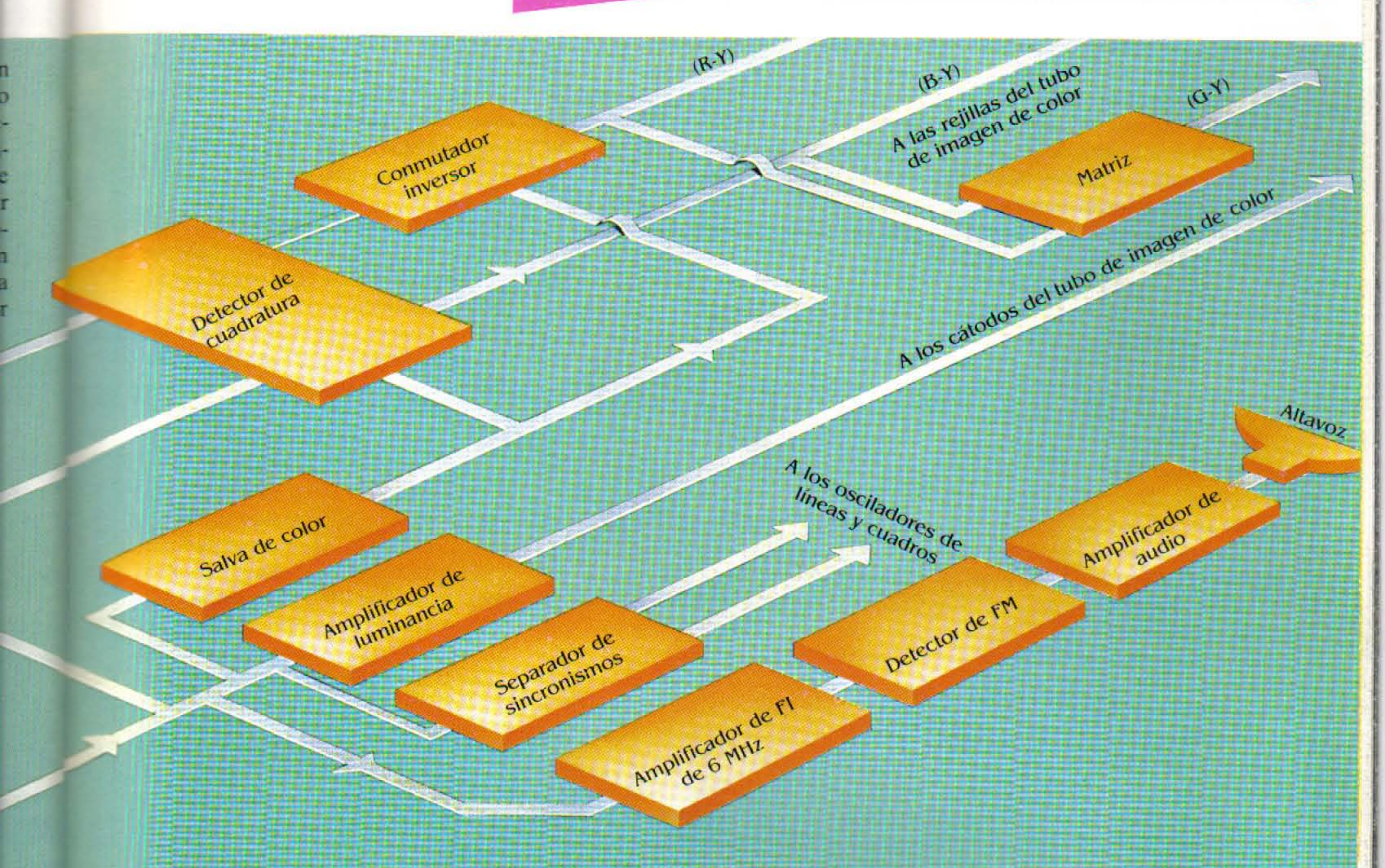
La señal compuesta de video constituye una especie de solución a medio camino entre la señal normal de televisión comercial y la entrada RGB. Sin embargo es mucho más aconsejable que la señal normal de televisión. Ello se debe que esta última señal sufre más procesos de modulación, conversión de frecuencia y amplificación hasta que se convierte en una imagen y es evidente que cuantos más procesos sufre una señal, más probable es que se vea contaminada por el ruido y la distorsión. En el modo RGB, la señal de sincronismo va por separado (en otra patilla del conector).

ANCHO DE BANDA

Existen unas razones técnicas interesantes aunque algo complicadas acerca del por qué un televisor no puede dar tan buena resolución como un monitor. Por ejemplo, un televisor no puede manejar información que requiera un ancho de banda en frecuencia superior a los 5,5 MHz. Con este ancho de banda se puede conseguir una presentación en pantalla de 40 columnas, pero para una presentación de 80 columnas que sea satisfactoria hace falta un ancho de banda superior a los 10 MHz.

Los monitores de video están especialmente diseñados para aceptar este ancho de banda. En casi todos los monitores disponibles en el mercado de ordenadores domésticos, la entrada prevista es la señal de video compuesto, por lo que no hace falta el sintonizador o el modulador que requieren los televisores. Debido a que necesita mucho menos procesamiento, la señal de vídeo compuesto de un ordenador aparece normalmente mucho más «limpia».

Cuando se habla de una «buena» imagen en la pantalla, normalmente se está haciendo referencia a la resolu-



ción y al color. Por desgracia ambas cosas no siempre son compatibles. Una pantalla de blanco y negro siempre tendrá una mejor definición de imagen que una pantalla en color. En una pantalla monocroma sólo se requiere un punto de fósforo para definir un punto de la imagen, mientras que se requieren tres puntos del mismo tamaño para definir ese mismo punto en una pantalla de color.

En una pantalla normal de cualquier clase hay unos 360.000 puntos de fósforo. En el caso de una pantalla de color se requerirían 120.000 de dichos puntos para el rojo, 120.000 para el verde y 120.000 para el azul. Esto significa que sólo hay disponibles 120.000 grupos de tres puntos para formar la imagen en una pantalla de color. Por el contrario una pantalla monocroma tiene 360.000 puntos disponibles para crear una imagen.

Todo el que haya trabajado con ambos tipos de pantalla, la monocroma y la de color, reconocerá inmediatamente la diferencia. El monitor en blanco y negro es mucho más nítido y claro que el de color, el cual presenta una imagen comparativamente más borrosa.

En muchas aplicaciones especializadas la elección natural será el monitor monocromo. El color normalmente elegido para estas aplicaciones —diseño asistido por ordenador o procesado de textos, por ejemplo— es el verde. Está considerado como el color más fácil para la vista y el menos cansado.

En cuanto a su peligrosidad para la salud, la inmensa mayoría de los informes que han sido publicados sobre el tema, declaran que el tubo de rayos catódicos es absolutamente inofensivo para los ojos. Normalmente se suelen oír quejas debidas a otras condiciones de trabajo, a colores de pantalla inadecuados, reverberaciones de la pantalla, distracción debida a otros focos luminosos y a veces efectos psicológicos. Sin embargo la pantalla de un televisor presenta un ligero centelleo y además suele estar hecha de un material reflectante que con frecuencia hace aparecer imágenes indeseadas y causantes de distracción. Esta es una de las razones por las que el uso de un televisor para un trabajo serio, que requiera una gran concentración durante largos períodos de tiempo, puede causar jaquecas.

Para producir una imagen con la misma resolución que las de la televisión o el vídeo ordinarios, nuestros ordenadores tendrían que controlar individualmente cada punto de fósforo de la pantalla. Y suponiendo que cada *pixel* necesite un byte de memoria, se necesitaría medio Megabyte para controlar toda la pantalla. Obviamente esto está fuera de lugar aquí, ya que

nos estamos ocupando de pequeños ordenadores domésticos y esa cantidad de memoria representa un tremendo gasto incluso en aplicaciones de ordenadores más grandes.

La mayoría de los micros domésticos disponen de aproximadamente 1K de memoria para dividir la pantalla entre 800 y 1000 cuadrados, típicamente 24 líneas por 40 columnas. Otros ordenadores te permiten elegir entre baja resolución y alta resolución, dependiendo de la forma de presentación seleccionada. Hay que observar sin embargo que esto no tiene nada que ver con la pantalla que utilices. Las limitaciones en la definición y la resolución de la pantalla vienen impuestas por el ordenador y no por el tubo de rayos catódicos.

Cada uno de estos aproximadamente 900 cuadrados es una casilla individual que puede contener un carácter. El ordenador contiene un conjunto de caracteres en ROM en el que corresponde un número del código ASCII para cada carácter. Cada carácter ocupa un cuadrado al ser presentado en la pantalla. De esta forma se puede almacenar un carácter en un solo byte, consiguiéndose un enorme ahorro de memoria.

HACIENDO UNA ELECCION

Los dos factores más importantes a tener en cuenta a la hora de comprar una pantalla nueva son el tipo de ordenador y las futuras aplicaciones. A pesar de ser un elemento importante dentro del sistema, la pantalla siempre ha de estar supeditada al ordenador. Si las capacidades gráficas del ordenador son pobres, seguirán siéndolo independientemente de lo buena que sea la pantalla. Siempre es el ordenador el que controla la pantalla y es importante que ésta se compre con arreglo a las características del ordenador.

También es importante tener en cuenta el uso a que se va a destinar el ordenador. En las aplicaciones en las que nunca va a ser necesario el color, la elección obligada es un monitor de blanco y negro. Si es esencial una imagen de alta calidad en colores, se requerirá un monitor con entrada RGB, si bien incluso para la mayoría de los usuarios profesionales de micros bastará con una entrada de vídeo compuesto. Para todos los poseedores de micros, con excepción de los de más alto precio, la elección entre un monitor de entrada RGB o uno de entrada de vídeo compuesto tendrá un interés puramente académico ya que se construyen muy pocos micros con salida RGB.

Para los juegos, la programación «recreativa» en general hecha en casa y otros usos domésticos de los ordenadores, bastará con un televisor ordinario. Aunque muchos juegos resultarían mucho mejor sobre pantallas de alta resolución, otros han sido escritos con conocimiento pleno de que sus gráficos sólo se iban a representar en un televisor ordinario. Ten presente además que debido a que los monitores se fabrican principalmente para aplicaciones comerciales, es frecuente que no incluyan características tales como el sonido.

Además, aunque su electrónica con frecuencia es más sencilla que la de un televisor, los monitores no son necesariamente más baratos que los televisores de un tamaño semejante. En lo que se refiere a pantallas, lo más grande no es necesariamente lo mejor. En muchos casos incluso podría ser mucho peor. Al aumentar o disminuir las dimensiones de la pantalla, lo que ocurre es que también aumenta o disminuye el tamaño de los *pixels* individuales. Si por ejemplo el ordenador produce 960 *pixels*, ése será el número de ellos que aparezcan en pantalla con independencia de lo grande o pequeña que ésta sea. Normalmente no se produce un aumento de la cantidad de información disponible en la pantalla al aumentar el tamaño de la misma, y en muchos casos lo que sucede es que una pantalla grande hace que ciertos contornos sean aún más destacados. Por cierto, recuerda que el tamaño de una pantalla se mide a lo largo de la diagonal de la misma, no según su anchura o su altura.

No se recomiendan los televisores viejos o los de muy bajo precio. Muchas veces lo que no parecería en absoluto un problema serio en una imagen de un programa emitido por una emisora de televisión comercial, puede convertirse en algo desastroso cuando el televisor se utiliza con un ordenador.

Por ejemplo el sobrebarrido, que recorta los lados de la parte superior e inferior de la pantalla, puede no parecer demasiado serio en la presentación de imágenes televisivas, pero podría ocurrir que en el caso de utilizar un ordenador no apareciera en pantalla alguna información útil. Otro defecto frecuente de los televisores viejos es el que ocurre cuando un color se presenta con mucho más intensidad que los otros, produciendo un efecto de emborronamiento que puede hacer que las letras de la pantalla resulten muy difíciles de leer. Las reparaciones para eliminar este tipo de defectos suelen ser normalmente muy caras. Con frecuencia es más barato comprar un televisor mejor... o un monitor.

CARACTERISTICAS DE LA VISUALIZACION

Tanto en el caso de un televisor como si tienes un monitor de video, puedes hacer varias cosas para mejorar las condiciones de visualización:

Puedes reducir la fatiga de la vista utilizando caracteres con colores claros contra un fondo negro. Para mejorar la definición utiliza los controles de brillo y contraste hasta lograr el ajuste óptimo. Los halos de color se pueden eliminar normalmente bajando la señal de color o eliminándola del todo; al fin y al cabo no necesitas el color para el procesamiento de textos.

Intenta situar tu mesa de trabajo, o el televisor o monitor, de forma que la pantalla no refleje la luz que entra por la ventana o la de alguna lámpara. En algunos casos puede ayudar el llevar ropas de tonos oscuros. Puedes minimizar aún más los efectos de las reflexiones teniendo la pantalla de forma que tengas que mirar ligeramente hacia abajo en vez de tenerla justamente enfrente de tus ojos o un poco más alta.

LOS MEJORES DE INPUT COMMODORE

PUESTO	TITULO	PORCENTAJE
1.°	Commando	25,6 %
2.°	Rambo	13,5 %
3.°	Kung Fu master	12,6 %
4.°	Skyfox	12,6 %
5.°	Hardball	7,6 %
6.°	Desert fox	6,3 %
7.°	The Dambusters	5,8 %
8.°	Misión imposible	5,7 %
9.°	Winter Games	5,4 %
10.°	Back to the future	4,9 %
		100 %

Para la confección de esta relación únicamente se han tenido en cuenta las votaciones enviadas por nuestros lectores de acuerdo con la sección «Los Mejores de Input».

Mayo de 1986

SOFTACTUALIDAD

Esta sección de actualidad presenta algunas novedades mundiales de *soft.*, aunque esto no quiere decir que se encuentren en nuestro mercado o vayan a llegar a él.

MODEL BUILDER, EL CONSTRUCTOR DE PROTOTIPOS

Siguiendo con la saga de los programas constructores de pistas de carreras automovilísticas, **Thorn Emi** acaba de sacar un fantástico constructor de coches, que comienza desde la elección del motor en 6,8 y ... hasta 12 cilindros en V, incorporación de turbo, etc.
Tras la instalación de la carroceria accedes a la carrera, en la que se incluyen circuitos urbanos (con vistas a las torres gemelas de Nueva York), campo a través, playas, las 12 millas de **Daytona**, etc... Lástima que la idea de las carreras de coches haya sido tan aprovechada ya.



CRIMEN Y CASTIGO, QUIEN LA HACE LA PAGA

Tomando el título de la célebre novela de **Dostoiveski**, **Bamtham** e **Imagir** distribuyen en Europa un



sorprendente juego que pone a prueba la justicia y ecuanimidad de un juez controlado por ordenador, americano por supuesto. El juez de una pequeña ciudad de Carolina del Norte, tendrá que examinar detenidamente una serie de delitos, con sus circunstancias agravantes, el historial del detenido y decidir por fín hacia que lado se inclinará la balanza de la justicia.
Este es un programa de estrategia, semiconversacional, de excelente nivel gráfico que sorprende pantalla tras pantalla.

★★★★★★★★★

LITTLE COMPUTER PEOPLE, ALGUIEN VIVE EN MI ORDENADOR

Hay seres que viven en el lugar más oculto que te puedas imaginar. Si!, eso parece pensar la casa **Activision**, que ha lanzado un nuevo programa que gira en torno a estos diminutos personajes. La pantalla muestra la casita de nuestro simpático inquilino, su perro, sus comodidades domésticas, su gato y así observamos sus tranquilas costumbres.
Como todo ser normal, el personaje debe mantener un riguroso ritmo de vida en lo que a ocio, alimentación, y reposo se refiere. Podrás ver con él la televisión o escuchar juntos sinfonias de **Beethoven**, incluso preparar la comida del perro. Pero el colmo de los colmos es que puedes



manejar desde tu ordenador el microordenador de tu inquieto amigo.
El mes que viene comentaremos este programa con mayor detenimiento.

★★★★★★★★★

PRINT MASTER

Estamos sin duda en medio de una época de esplendor en el uso de las impresoras. Primero fué **Print Shop**, luego el **Newsroom** y ahora le llega el turno del **Print Master** al **C-64**, que se lanzó originalmente en versión para el **Amiga** y el **520** de **Atari**.
Dispone de una gama de posibilidades muy variadas, con siete



tipos de caligrafía totalmente nuevos y una mayor facilidad de composición de la tarjeta (texto en bandera, etc.).
Otras opciones te permitirán visualizar calendarios, tarjetas de visita y, lo más importante y novedoso con respecto a sus predecesores, ver como quedará el resultado de tu labor.

YIE AR KUNG FU

Yie ar Kung fu, un nuevo «karate» como su nombre indica. Tienes que luchar contra diez contrincantes. Ojo, cada uno tiene su truco y vamos a tratar de explicarte los que hemos hallado. El primer *ninja* no lleva armas de ninguna clase, contra él basta pegar brincos y tratar de arrearle por la espalda. Esta táctica es también válida con el oponente siguiente, que esta vez te lanza estrellas puntiagudas. En este caso saltar, golpear su espalda y volver a saltar es la única alternativa viable si quieres agotar sus diez puntos de energía, sin que él te golpee. Acabar con cualquier adversario sin que te quite ningún punto de energía te dará 40.000 puntos más. El jugador siguiente es muy peligroso. Cuidado, con el palo que lleva entre manos, y que maneja con una destreza asombrosa, hay una estrategia que no falla: agacharse.

KINETIC, INTERESANTE ROMPECABEZAS

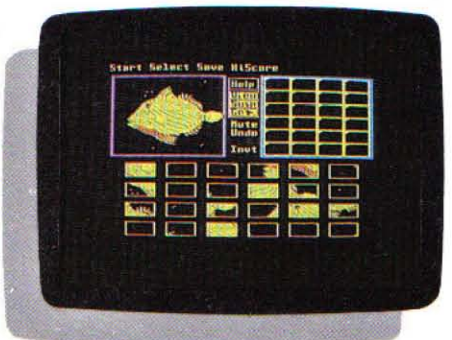
Kinetic es un juego de resolución de rompecabezas en movimiento, muy difícil de resolver, como demuestran las fotografías, cuyas figuras son harto complicadas. Es una muestra mas del *Free Soft*, o *software* que pagas únicamente si te gusta.
Es un programa muy flexible con multitud de figuras diferentes, en las que puedes aumentar el número de piezas que componen el rompecabezas y hasta realizar dificilísimas variaciones de colores que hacen prácticamente imposible controlarlo.



MASQUERADE

No, no vayas a creer que este es el juego que corresponde al popular serial de televisión. **Masquerade** es un juego antiguo, quizás con mas de dos años, pero que los amantes de los juegos conversacionales toman como muestra de algo que no ha sido superado aún. La calidad gráfica y la dificultad del juego han hecho del mismo una prueba de fuego de la capacidad del jugador.
La trama es harto sencilla, como presenciador de un asesinato has de lograr escapar del escenario del crimen y lograr demostrar tu inocencia, hallando el móvil del crimen y, claro está, al asesino.







LOS MEJORES DE INPUT

En la revista de Abril, y debido a un error de montaje, apareció publicada la lista correspondiente a INPUT-SINCLAIR. A fin de que podáis seguir la marcha de vuestros favoritos reproducimos aquí la relación correcta.

PUESTO	TITULO	PORCENTAJE
1.°	Skyfox	23,7 %
2.°	Rambo	14,5 %
3.°	Winter Games	13,7 %
4.°	Commando	9,1 %
5.°	Hardball	8,4 %
6.°	Karateka	8,4 %
7.°	Kung Fu master	6,5 %
8.°	La ley del Oeste	5,9 %
9.°	Pitstop II	5,3 %
10.°	Rock'n'lucha	4,5 %
Abril de 1986		100 %

CUIDADO CON LAS ROBUSTAS AMAZONAS

El tema de **La mujer Amazona** es un clásico en los juegos de «arcadia»: luchas violentas en cada recodo del camino.
Este programa recuerda vagamente a **Fighting Warrior**, pero esta vez eres una fémina perdida en medio de las temibles **Amazonas** y debes recorrer tu largo periplo, hasta encontrar a tu querida hijita.
Como es lógico, en la aventura tendrás que luchar contra otras mujeres (en este juego no hay personal masculino). Los golpes permitidos son los mismos que en **Fighting**, aunque la acción está quizás menos trabajada y la velocidad a la que luchas es más lenta. En la parte superior de la pantalla aparece una especie de rádar, que te indica cuáles son las mujeres que interceptan el camino, así como el sentido y la altura a la que se aproximan unas flechas tiradas por seres invisibles.
Este es sin duda un juego desarrollado originalmente para el microordenador **Spectrum**, y esto se nota en la versión para **Commodore** y no se le ha sacado demasiado rendimiento a las mayores prestaciones gráficas de este ordenador. No obstante sigue siendo un juego notablemente entretenido.







DATOS GENERALES

TITULO Legend of the amazon women

FABRICANTE U.S. Gold

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Lucha entre féminas

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	7
INTERES	8
GRAFICOS	7
COLOR	7
SONIDO	7
TOTAL	36

****

KANE

Estamos ante un juego de la nueva serie de calidad y bajo precio de compra.
Kane es el nombre de una ciudad del lejano oeste y tú eres el *marshall* **McGraw** —el alguacil de pistola fácil— con ansias de llegar a ella.
Una serie de cosas tratarán de impedirlo, aunque también tienes que lograr la paz con los indios.
El juego está dividido en cuatro fases y puedes optar por jugarlas todas, quedando eliminado a medida que falles, o practicarlas independientemente.
Llegas a la aventura cabalgando por las praderas a lomos de tu fiel e inseparable caballo. Más tarde desmontas y alcanzas corriendo el valle en el que se encuentra el poblado indio (la perfección de movimiento en la carrera del vaquero recuerda al agente de **Misión Imposible**), lugar en el que tendrás la posibilidad de demostrar tu habilidad en la caza de patos con el arco y las flechas. Tu destreza te dará valiosos puntos para la paz, que habrán de traducirse en la firma de un tratado con el «gran jefe blanco» en Washington.
Como en las películas, siempre hay problemas derivados de la insaciable ambición de riquezas del hombre blanco. El inevitable trazado de la nueva línea férrea, que trae la «civilización» y pingües beneficios a los ferroviarios, pasa por las tierras que los indios habitaron durante generaciones. Aquí tienes oportunidad de galopar a través de una carrera de obstáculos, en la que

DATOS GENERALES

TITULO Kane

FABRICANTE Mastertronic

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Habilidad

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	**9**
INTERES	**8**
GRAFICOS	**8**
COLOR	**7,5**
SONIDO	**8**
TOTAL	**40,5**





tu oponente es el caballo de hierro. Antes puedes practicar por el desierto, saltando rocas y matorrales, pero cuidado, no es tan fácil como parecía.
Cuando adelantes al tren las cosas serán más favorables para el tratado. Y por fin en el poblado de los rostros pálidos, donde habrás de mantener un tiroteo con los indeseables que tratarán acabar contigo. Debes acabar con los seis o ellos lo harán contigo. Cuando alguien cae, se oye un grito inhumano de dolor. Cuando se te acaben las balas, tendrás que recargar el revólver saliendo de la pantalla.
Los bandidos aparecen en lugares insospechados, una puerta que se abre, una ventana o incluso el tejado.
La calidad de los *scrollings* y el movimiento de los *sprites* hacen de **Kane** un juego enormemente atractivo, aunque sea difícil empezar a dominarlo antes de un buen rato, dependiendo, claro está de la destreza que tengas en el manejo del *joystick*.

****

VIAJERO EN EL TIEMPO

Basado en la película del mismo título, **Regreso al futuro** es un programa en el que tu asumes la personalidad de **Marty**. Llegas al pasado y has de conseguir poner en marcha un romance entre dos casi adolescentes, para de este modo conseguir llegar a existir. El objetivo es claro, **Lorraine** y **George** —tus futuros padres— deberán besarse, después podrás regresar al futuro, tu época.

Tienes que utilizar diversos elementos, que surten algún efecto particular sobre los personajes y deberás descubrir cual hace qué. Podrás localizarlos en diferentes ubicaciones, tales como la cafetería, la casa de **Doc** o la discoteca.
El control de **Marty** se realiza mediante el *joystick*, con la ayuda de unas figurillas que aparecen en la pantalla. Entre ellas está un patinete que le sirve para desplazarse a mayor velocidad.
La pantalla se divide en cuatro áreas de trabajo (ventanas). La ventana principal muestra el desfile de los

DATOS GENERALES

TITULO Regreso al futuro

FABRICANTE Activision

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Aventura cinematográfica

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	**9**
INTERES	**7**
GRAFICOS	**8**
COLOR	**7,5**
SONIDO	**8**
TOTAL	**39,5**







distintos escenarios, al principio es la calle principal del pueblo. Con el *joystick* abres las puertas de los lugares a los que deseas acceder. La



ventana derecha es ocupada por una foto digitalizada de la familia **Marty**, que desaparece en la medida en que vas errando. Pero cuidado, si la foto llega a esfumarse no llegarás a ver este mundo.

A la izquierda observamos una foto similar, esta vez del propio **Marty**, que tiene una función similar a la del retrato de los futuribles progenitores. Aquí sucede que cuando logras tener juntos a **George** y **Lorraine** has de ir a la biblioteca y recoger el libro de



poesía romantica, que hará que ella le mire con buenos ojos.

Debajo de la ventana central se reflejan los semblantes de **Doc**, **Biff**, **George** y **Lorraine** (todos presentes), con el fin de que podamos conocer en qué momentos están juntos entre sí. El parpadeo de las imágenes denuncia esta circunstancia.

Inmediatamente debajo se ha dispuesto un reloj digital, que facilita conocer el tiempo que llevas jugando.

Los objetos anteriormente mencionados son representados en la parte superior de la pantalla.

Podemos encontrar una taza de café, utilizable para detener a **Biff** durante treinta segundos. Sin embargo, el casco espacial hace que **George**, tu futuro padre, te siga. Por su parte, la guitarra sirve para cautivar (congelar) a **Lorraine**.

No olvides que cuando los retratos se tornan verdes, tu dispones algún tipo de influencia sobre el personaje.

La calidad gráfica no deja nada que desear, resaltando la suavidad lograda en el desplazamiento de los fondos (*scrooling*).

El sonido, música incluida es igualmente loable.

Es un juego que sin duda dará mucho que hablar.

EL TUNEL DEL TIEMPO

En esta aventura te conviertes en un personaje poco habitual, eres ni más ni menos que un *gnomo*. Eso sí, con serias aspiraciones a ser el rey de los tuyos. Pero dice un antiguo proverbio que no hay atajo sin trabajo y el que te corresponde es hallar las siete páginas perdidas de un antiquísimo manuscrito sagrado, que fueron esparcidas por el tiempo. Tu juventud, apenas ciento quince



años, será un envidiable aliado en la tarea.

Además eres lo suficientemente habilidoso como para construir una máquina que te servirá para viajar por el tiempo. En cada época a la que viajes encontrarás parte del manuscrito.

El primer salto puede muy bien ser a la Edad de Piedra, hace casi más de diez mil años. En una cueva encuentras un misterioso y útil bastón, que has de aprender a manejar.

El agujero negro, situado a diez mil años de nuestra era, es el lugar al que puedes llegar una vez conseguidas todas las páginas.

La fiebre del oro californiano, a mediados del pasado siglo, te transporta al lejano oeste. Un caballo te permite encontrar otra página en alguna cueva de las tres existentes.

La Grecia mitológica tiene cinco pantallas que te permiten deambular por el templo. Siempre hay un ladrillo suelto, que te ofrece algunas posibilidades. Ten cuidado con la

DATOS GENERALES

TITULO Time Tunel

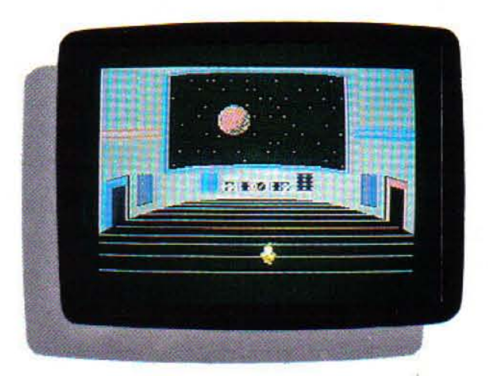
FABRICANTE U.S. Gold

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Viaje por el tiempo

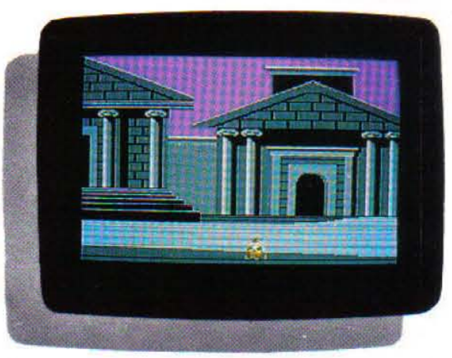
CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	9,5
INTERES	8,5
GRAFICOS	8,5
COLOR	8
SONIDO	7
TOTAL	41,5



medusa, porque sus proyectiles pueden devolverte a casa por la vía rápida.

En la mágica Persia de hace tres mil años, debes seguir a una misteriosa mujer que te ayudará a cumplir la misión de este tiempo. Hay cajas



fuertes y piedras voladoras.

Salem también guarda su hoja del manuscrito. Esta ciudad americana, famosa por su quema de brujas, está poblada únicamente por murciélagos que, si te tocan, pueden hacerte regresar al punto de partida. Aunque también los puedes abatir con disparos. Investiga lo que ocurre en el interior de una farmacia o una carnicería.

La nave intergaláctica tiene un complejo panel de control, del que has de recoger el signo más, que será de ayuda cuando recorras la nave.

La originalidad de este juego, junto a la enorme cantidad de pantallas diferentes con las que cuenta, proporciona mucho rato de variado entretenimiento. En realidad puede calificarse de auténtico rompecabezas.



A NUESTROS NUEVOS LECTORES

En las páginas centrales de la revista encontrarás la sección «Programación de juegos» que se compone de una serie de artículos coleccionables que continúan mes tras mes.

Como la paginación de estos artículos es siempre correlativa con la del mes anterior apreciarás que no se corresponde con la del resto de la revista, pudiendo parecer a los más despistados que faltan páginas o que se trata de un error de encuadernación.

CONCURSO INPUT/ANAYA MULTIMEDIA

¡DESCUBRE AL ASESINO!

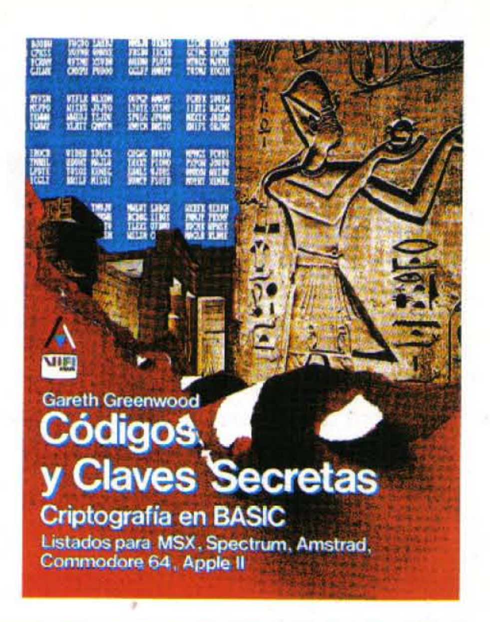
BASES:

Durante tres números sucesivos irán apareciendo en **INPUT** las tres partes del relato «El caso del anciano asesinado» junto con los mensajes cifrados que constituyen las respuestas parciales a la solución del crimen. Cada mes, quienes logréis descifrarlos participaréis en un sorteo consistente en 10 lotes de libros por un valor de 15.000 ptas., a elegir del fondo editorial de **Anaya Multimedia**, y una suscripción gratuita por un año a **INPUT**. Habrá tres sorteos, de tal forma que no será necesario haber descifrado el enigma del mes anterior para optar al premio.

Un fabuloso premio será la guinda final de este pastel. El descubridor de las motivaciones del crimen visitará al próximo **PCW Show** a celebrar en Londres. En caso de haber más de un acertante, recurriremos a la inevitable fórmula de sorteo.

La solución del primer mensaje cifrado deberéis enviarla, junto con vuestros datos personales, a la Redacción de **INPUT** antes del 15 de junio.

Las decisiones del jurado serán inapelables, dándose en las páginas de **INPUT** cumplida cuenta de la marcha del concurso.



*Nota: El libro de reciente aparición «Códigos y claves secretas», de **Anaya**, contiene todos los programas útiles para esta labor. Aunque si has trabajado para los servicios de inteligencia de alguna potencia no te será de gran utilidad.*



EL CASO DEL ANCIANO ASESINADO

La muerte es una forma de desorden; cae sobre la vida como el hacha sobre el tronco y quiebra siempre la sintaxis que después es preciso recomponer con dolor, a veces con recuerdos y, casi siempre, con notables dosis de olvido.

En otras ocasiones, la muerte es una estupidez o el resultado de varias estupideces hábilmente engarzadas por el azar o por la ambición. Investigando este tipo de muerte me gano la vida.

Aquella tarde, la primavera se había desplomado sobre las calles con una bola de fuego en sus entrañas. Me llamaron por teléfono y tuve que atravesar la ciudad en un momento especialmente difícil. Los conductores, todavía con sus ropas de invierno, se asaban de calor dentro de sus coches, lo que les ponía particularmente agresivos. Los semáforos, por el contrario, parecían haber sido tratados con tranquilizantes y pasaban del rojo al ámbar y del ámbar al verde con una parsimonia que desesperaba todavía más a los sufridos conductores.

Al fin llegué a una zona residencial y subí a un piso de cinco estrellas que acababan de abandonar los bomberos. Mi ayudante ya estaba allí, intentando preservar las escasas huellas que se habían resistido a la violencia de éstos. El incendio había sido de escasa magnitud, pero había bastado para asfixiar a un pobre viejo que permanecía atado a una de las sillas del salón. Una caja fuerte, hasta entonces oculta por un cuadro, aparecía abierta y desvalijada. Los bordes de la puerta de acero presentaban señales de violencia. Anoté en uno de mis códigos cifrados:

```
QE GZM HUAXQZOUM RUZSUPM BGQE EQ FDMFM P
Q GZ FUBA PQ OMVM UZMOOQEUNXQ M OGMXCGUQ
D FUBA PQ BMXMZOM
```

Al parecer, el incendio no había sido intencionado. Según los expertos, el origen fue una colilla que, tras resbalar del cenicero, había ido a caer al pie de una cortina originando un incendio de pequeñas dimensiones, pero con un humo lo suficientemente espeso para asfixiar a un anciano y hacer llegar a los bomberos.

Juan José Millas

CARRERA CONTRA EL RELOJ

Nada más entrar en funcionamiento, este programa emite una orden vocalizada en inglés: «*V-8, regrese a la base*».
Se supone que te encuentras en la etapa posterior a un holocausto nuclear y dispones de un tiempo limitado para alcanzar el seguro garage antes de que la radiación te convierta en un mutante.
La pantalla se divide en tres áreas,



donde la franja superior es la que te permite sacarle partido al juego. Es una trabajada vista aérea del vehículo que conduces a medida que avanzas por el camino. El *scrolling* del terreno y los detalles gráficos que lo componen son sensacionales. El control del coche —a través del *joystick*— es muy sensible, por lo que no puedes lanzarte a una loca carrera contra el tiempo, sino que debes aminorar la velocidad en las curvas.
Detalle curioso, los movimientos del *joystick* corresponden a la misma dirección seguida por el bólido en la pantalla, es decir, al comienzo del juego avanzas si el *joystick* es desviado hacia la izquierda (no hacia adelante como en otros juegos).

DATOS GENERALES

TITULO The Last V-8

FABRICANTE Mastertronic

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Holocausto postnuclear

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	**9**
INTERES	**8,5**
GRAFICOS	**9,5**
COLOR	**9**
SONIDO	**8**
TOTAL	**44**



Una vez que has recorrido zonas rodeadas por campo, llegas a un túnel que te introduce en la ciudad, donde el manejo del volante y el acelerador deber ser más cuidadoso.
En el área central de la pantalla aparecen los instrumentos de control, desde el ordenador de abordo hasta el cuentarrevoluciones y velocímetro, medidor del tiempo que falta, turbo



o carburante.
La última franja es decorativa y en ella aparece una vista lateral del V-8.
El juego resulta muy estimulante por la gran calidad de los contenidos gráficos y, sobre todo, la carrera contra el tiempo. Asimismo, la música de fondo suena incesantemente.
El último V-8 es otro de los juegos de **Mastertronic** orientado a luchar contra la piratería utilizando como arma los precios bajos, pero sin que ello suponga que se haya sacrificado la calidad.

****



EL ZOCO

Deseo cambiar juegos para CBM-64, también compro.

Jordi Duatis Juárez.
Montserrat, 18.
Teléf. (93) 792 19 90.
Arenys de Mar (Barcelona).

Vendo Datassette para Commodore 64. Fué comprado el 17-1-85, está nuevo. Precio a convenir.

José Ramón Díaz González.
Río Requejada, 2 - 1° izqda.
Teléf. (985) 57 28 31.
Avilés (Asturias).

Cambio programas para Commodore-64. Tengo cassette. Me interesa todo tipo de programas: juegos, educativos, aplicaciones... Prometo contestar.

José Manuel Barral.
Pza. de los Dolores,8 - 6° A.
21002 Huelva.

Vendo VIC-20, ampliación 16K y algún juego, por comprar un C-64; precio a convenir.

Federico Arias.
Teléfs. 715 45 18 - 241 49 87.
(Por las tardes).
Madrid.

Vendo Vic-20 casi nuevo más el juego «Centipede» (cinta), todo por 15.000 ptas.

Joaquín Cid Morán.
Rda. de San Pablo 29, S/A.
Teléf. (93) 241 17 41.
08015 Barcelona.

Cambio toda clase de programas para el C-64. Poseo cassette y disco.

Manuel González Lombardía.
Gabriel y Galán, 4 - 2° A.
Peñaranda de Bracamonte
(Salamanca).

Vendo VIC-20 por 20.000 ptas. Está incluido en el precio ampliación 8K. Muchas revistas, programas y más cosas.

Rafael Riera Trias.
Granados s/n.
Teléf. 564 17 03.
Montcada Reixach (Barcelona).

Intercambio programas en disco y cinta para el C-64.

Pablo Acinos.
Panaderos 14, 5° C.
34001 Palencia.

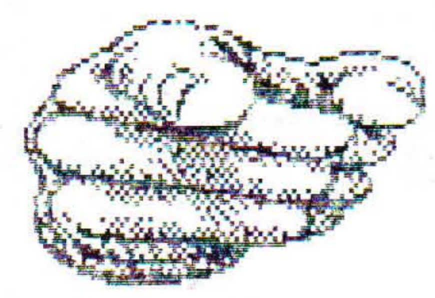
Desearía ponerme en contacto con usuarios de Commodore 64 en San Lorenzo del Escorial, para cambiar juegos, cartuchos, etc.

Teléf. 890 26 30.



Si alguien sabe construir un joystick sólo de botones, o sea, de teclado, o quiere intercambiar programas de juegos y/o utilidades (cinta), escribir a:

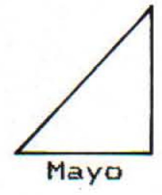
Juan Jurado Benimeli.
Eduardo Bosca 24-36.
46023 Valencia.

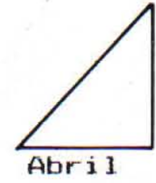
TU  PUEDES SER EL AFORTUNADO

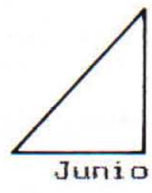
que reciba gratuitamente en su casa durante un ano todos los programas que sean lanzados por Erbe o U.S. Gold para los ordenadores Commodore. Podras participar en el sorteo con solo enviarnos esta pagina una vez que hayas pegado las cuatro esquinas coloreadas que apareceran en los tres numeros siguientes de INPUT. Ademas tienes que recortar de alguna caratula de un *cassette* los anagramas de Erbe y U.S. Gold y pegar en los lugares indicados.
La fecha tope de participacion es el 30 de julio, publicandose en INPUT de septiembre el nombre del ganador.

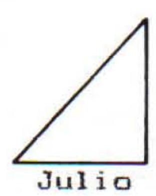
Datos del concursante:
Nombre:...
1er. Apellido:......................................
2o. Apellido:.......................................
Ano de nacimiento:...........
Direccion:........................... No.:......
Ciudad.................... Provincia:............
Dto.Postal:.............. Telefono:.............

Nota: Si bien esta pagina puede ser fotocopiada o copiada a maquina, no entraran en el concurso aquellas a las que falte alguno de los seis recortes exigidos o que estos sean copia del original.


Mayo


Abril


Junio


Julio

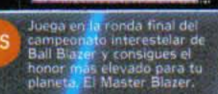




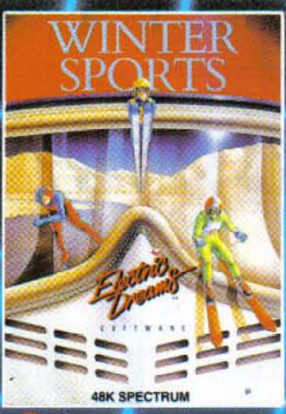
Mayo

ACTIVISION
HOME COMPUTER SOFTWARE





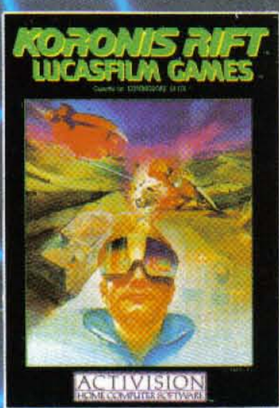
C.S Juega en la ronda final del campeonato interestelar de Ball Blazer y consigues el honor más elevado para tu planeta. El Master Blazer.



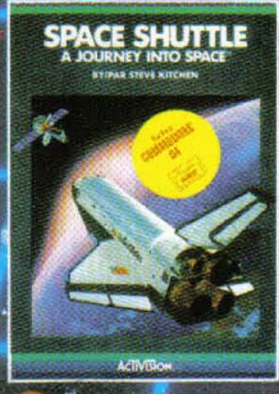
S.A Si no puedes practicar tus deportes de invierno favoritos te ofrecemos oportunidad de hacerlo con tu ordenador. Ocho competiciones distintas te esperan. Apresurate a equiparte y disfruta con el descenso, slalom, salto ski etc.



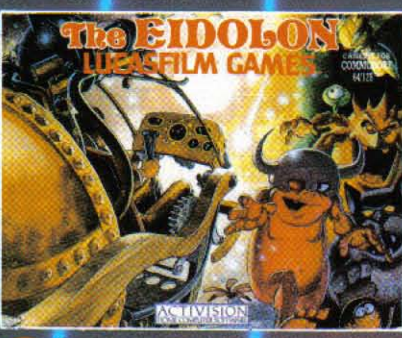
CSA Ya puedes disfrutar en tu ordenador con la pelicula que nos ofrece Steven Spielberg. No esperes ir al cine. Incluye la banda sonora original.



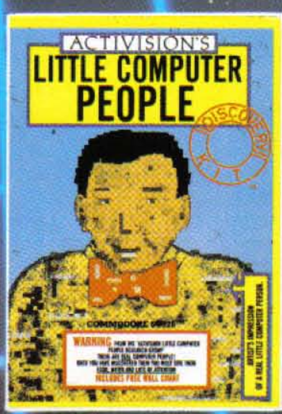
C Encuentra un tesoro en un remoto sistema estelar, pero ¡cuidado! con sus guardianes. Si quieres sobrevivir elabora un cuidadoso plan. La recompensa está fuera de este mundo.



C.S. El más reciente avance técnico. Vd. puede realizar una jornada completa dentro de una capsula espacial desde que se levanta de la superficie de la tierra y acude a un encuentro en el espacio hasta que aterriza nuevamente. Compruebe su habilidad.



C Descubre los secretos de Eidolon a traves de una misteriosa maquina del siglo XIX y sumergete en un mundo subterraneo poblado de curiosas criaturas mágicas y feroces dragones guardianes.



C Ya hemos descubierto a unos pequeños seres en el ordenador, ahora tienes que cuidar de no contradecirlos y que estén lo mas contentos posible. ¡Inténtalo!



S El laberinto mágico. Introdúcete en el movimiento tridimensional del planeta Newgama III y egcuentra las piezas necesarias para completar el robot.

... y sus clásicos GHOSTBUSTER. C.S.A.M. **DECATHLON DE ACTIVISION.** C **RIVER RAID.** C.S.M.

* ENDURO S PITFALL 2 C.S.M. y otros titulos más • **SPACE SHUTTLE.** C **DISIGNER PENCIL.** C **...etc.**

ENCUENTRALO EN LA DIVISION Online DE GALERIAS



Disponibles para:
COMMODORE C
SPECTRUM S
AMSTRAD A
MSX M

DELEGACION EN CATALUÑA:
C/París, 132 - 134
08011 Barcelona
Teléfono: 93 322 72 11

EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A:

Distribuido por:

PROEIN, S. A.
Velázquez, 10 - 28001 Madrid
Tels. 91/276 22 08/09